SCA Tool Security Audit

MASTER'S THESIS

André Rosenberger

Submitted on 17 March 2025

Friedrich-Alexander-Universität Erlangen-Nürnberg
Faculty of Engineering, Department Computer Science
Professorship for Open-Source Software

Supervisor:
Martin Wagner, M.Sc.
Prof. Dr. Dirk Riehle, M.B.A.

Friedrich-Alexander-Universität
Faculty of Engineering

Declaration of Originality

I confirm that I have written this thesis unaided and without using sources other than those listed and that this thesis has never been submitted to another examination authority and accepted as part of an examination achievement, neither in this form nor in a similar form. All content that was taken from a third party either verbatim or in substance has been acknowledged as such.

Erlangen, 17 March 2025

License

Erlangen, 17 March 2025

Abstract

A security audit of *SCA Tool Application*, its IT infrastructure's configuration, and the ISM within the organization *SCA Tool* was conducted. The objective was to establish a foundation for an ISMS and provide actionable recommendations for specific vulnerabilities.

During the vulnerability assessment, penetration testing tools were utilized, source code analyses were performed, and organizational vulnerabilities were identified through questionnaires. Based on existing ISMS frameworks, policies were developed for areas of *SCA Tool* requiring immediate action. Mitigation strategies for the identified vulnerabilities were formulated in accordance with these policies. In this process, technical documentation of the software modules in use, regulatory requirements, and the organizational context were considered.

This thesis has demonstrated the need for action regarding security in *SCA Tool*. It has highlighted the importance of best practices in secure software development and consistency in source code. Additionally, multiple layers of security are essential for a secure web application. Finally, further recommendations for the future enhancement of security for *SCA Tool* were provided.

Contents

List of Figures

List of Tables

Acronyms

ADP	Authorized Data Publisher
AiTM	Adversary-in-the-Middle
API	Application Programming Interface
AppSec	Application Security
ATO	Account Takeover
BCM	Business Continuity Management
BSI	Federal Office for Information Security
CA	Certificate Authority
CBC	Cipher Block Chaining
CD	Compact Disc
CLI	Command-Line Interface
CLOUD	Clarifying Lawful Overseas Use of Data
CNA	CVE Numbering Authority
CORS	Cross-Origin Resource Sharing
CPU	Central Processing Unit
CRA	Cyber Resilience Act
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
DAST	Dynamic Application Security Testing
DB	Database
DDoS	Distributed Denial of Service

DNS	Domain Name System
DoS	Denial of Service
DPA	Data Processing Agreement
DTLS	Datagram Transport Layer Security
E2E	End-to-End
EO	Executive Order
EU	European Union
FAU	Friedrich-Alexander-Universität Erlangen-Nürnberg
FISA	Foreign Intelligence Surveillance Act
GDPR	General Data Protection Regulation
HDD	Hard Disk Drive
HIBP	Have I Been Pwned
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IAM	Identity and Access Management
ICS	Industrial Control Systems
ID	Identifier
IdMS	Identity Management System
IoT	Internet of Things
IP	Internet Protocol
IS	Information Security
ISM	Information Security Management
ISMS	Information Security Management System
ISO	International Organization for Standardization
IT	Information Technology
JSON	JavaScript Object Notation
MFA	Multi-Factor Authentication
NDA	Non-Disclosure Agreement

NIST	National Institute of Standards and Technology
NSA	National Security Agency
NVD	National Vulnerability Database
OSS	Open Source Software
OWASP	Open Worldwide Application Security Project
PDCA	Plan-Do-Check-Act
PGP	Pretty Good Privacy
PHC	Password Hashing Competition
PII	Personally Identifiable Information
PIN	Personal Identification Number
PoC	Proof of Concept
PVE	Proxmox Virtual Environment
RRZE	Regionales Rechenzentrum Erlangen
SAST	Static Application Security Testing
SBOM	Software Bill of Materials
SHA-1	Secure Hash Algorithm 1
SIEM	Security Information and Event Management
S/MIME	Secure/Multipurpose Internet Mail Extensions
SMS	Short Message Service
SOC 2	System and Organization Controls 2
SOP	Same-Origin Policy
SSD	Solid-State-Drive
SSL	Secure Socket Layer
TLS	Transport Layer Security
ToS	Terms of Service
TOTP	Time-Based One-Time Password
UCC	Unified Communications and Collaboration
UI	User Interface
URL	Uniform Resource Locator

US	United States
USA	United States of America
VM	Virtual Machine
VPN	Virtual Private Network
WAF	Web Application Firewall
Wi-Fi	Wireless Fidelity
XSS	Cross-Site-Scripting
YAML	YAML Ain’t Markup Language
ZAP	Zed Attack Proxy

1 Introduction

On October 23, 2024, the Cyber Resilience Act (CRA) was adopted by the European Parliament and the Council of the European Union. It requires an appropriate level of cybersecurity from manufacturers of products with digital elements and must be implemented in its entirety by December 11, 2027 at the latest. *Annex I* of the CRA describes clear requirements that the product must meet, such as protection of the Information Security (IS) objectives confidentiality, integrity and availability, or limiting the attack surface. It also states that the manufacturer must identify and document vulnerabilities and regularly check the security of the product (European Parliament and Council of European Union, 2024).

The project *SCA Tool* of the *Professorship for Open-Source Software* at Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) is affected by this regulation and therefore an audit is to be concluded, in which IS, cybersecurity and data protection are to be evaluated. The assessment in this Master's thesis examines a wide array of subjects that are essential for *SCA Tool* and the development and operation of their product: *SCA Tool Application*, a TypeScript-based React web application, which aims at ensuring the proper utilization of Open Source Software (OSS) in products regarding licensing and vulnerabilities. They may include issues within the source code, organizational challenges, as well as regulatory considerations. Through this work, the groundwork for future assessments is established. With each iteration addressing various topics, *SCA Tool*, also referred to as organization, and *SCA Tool Application* attain a professional level of security. This audit occurs in several stages.

To create a basic understanding for the reader of this Master's thesis, information regarding the significance of audits and information security frameworks are provided in chapter 2.

During the initial phase, vulnerabilities and weaknesses are identified and evaluated for their potential impact on confidentiality, integrity, and availability. The identified issues, along with additional explanations, are listed in chapter 3.

In chapter 4, policies are introduced. These policies are based on established IS frameworks and the issues found in chapter 3. By adhering to these policies, the emergence of future issues and associated risks should be mitigated.

Based on the identified issues and the formulated policies, solutions for these issues are detailed in chapter 5. These solutions may involve both technical and organizational measures. The potential mitigations will be examined thoroughly to ensure comprehensive protection against the identified vulnerabilities.

In chapter 6, an evaluation is conducted to summarize how the requirements of this thesis were addressed by this work.

Conclusions about the security of *SCA Tool* and *SCA Tool Application* based on the issues, remediations and insights presented throughout this thesis are drawn in chapter 7.

Given the extensive nature of this topic, additional suggestions on the aspects to evaluate concerning security and compliance in future audits can be found in chapter 8.

2 Literature Review

To understand the context of a security audit, this chapter introduces security audits and Information Security Management System (ISMS), as well as frameworks supporting organizations in their Information Security Management (ISM).

2.1 ISMS and Security Audits

Brenner et al. (2024) recommend introducing an ISMS in order to ensure IS in an organization. An ISMS cannot be introduced without further ado, but must be adapted to each organization. Accordingly, the introduction of an ISMS is a strategic decision for the organization and is largely dependent on its needs and objectives. Based on the security requirements, the organizational processes and the structure of the organization, suitable policies, processes and procedures are implemented to ensure the IS objectives of confidentiality, integrity and availability. Policies show the formally defined directive of the organizational management. Processes are sets of interconnected resources and activities that transform inputs into outputs. Procedures then define how a process is executed. The mapping of the concepts to the individual levels is shown in figure 2.1. Due to the variety and complexity of these concepts, as well as the fact that they should be able to be implemented by alternating personnel, these concepts must be adequately documented in an ISMS. Based on these, measures for IS are to be taken. They do not have to guarantee a perfect level of protection, but must offer the highest possible level based on the requirements of the organization and within the scope of these requirements. The organization's requirements can change over time. Therefore, the process of planning, implementing, reviewing and improving the ISMS is an ongoing effort in order to establish a certain standard of security and to enhance its quality. This progression is known as the Deming cycle or the PDCA-cycle (*Plan-Do-Check-Act*). By applying the Deming methodology, the maturity level of the ISMS is improved over time and consolidated by standardization as shown in figure 2.2.

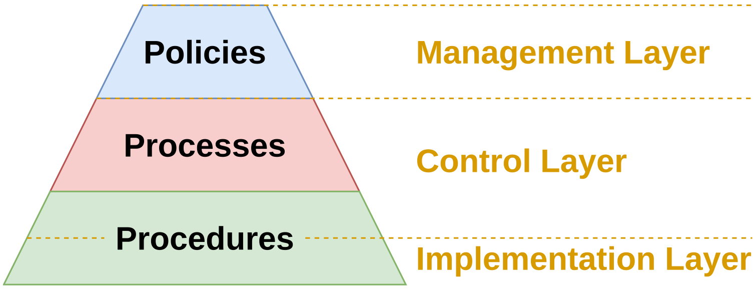


Figure 2.1: Concepts and Layers of an ISMS

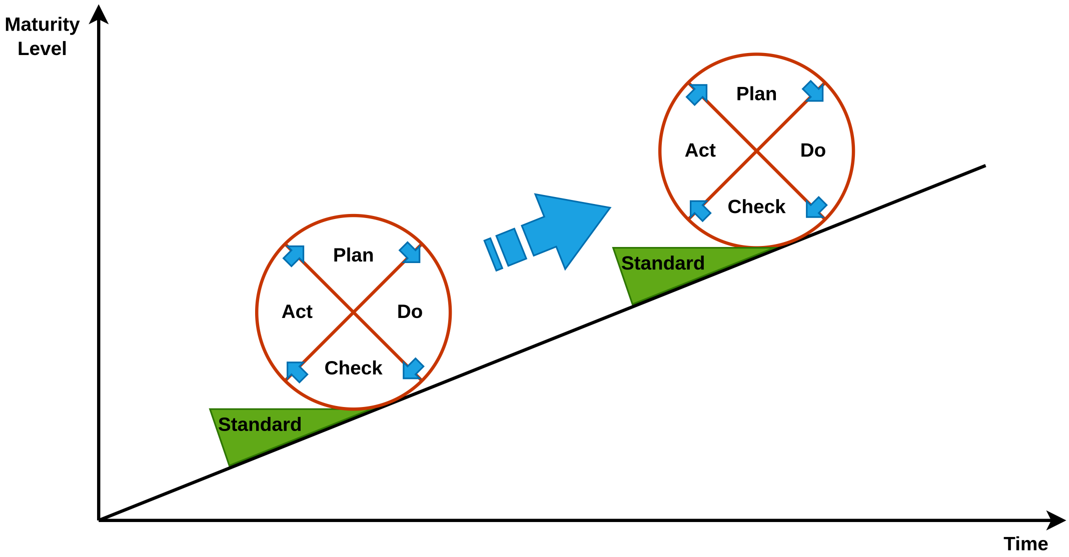


Figure 2.2: The PDCA-Cycle in Context of an ISMS

Various IS standards and norms exist, which help to set up an ISMS and firmly integrate it into the organization. Two renowned frameworks, the *ISO/IEC 27000* standard family and the *BSI IT-Grundschutz*, are introduced in section 2.2.

An organization should review its operational IT-processes by conducting security audits, which ensure the quality of the ISMS and compliance with legal and regulatory standards (Bruma, 2021). With table 2.1, Jadhav (2023, Table 1) shows a structured overview of the critical components reviewed during cybersecurity audits to ensure comprehensive protection across various domains. The table puts the audit into perspective and splits it into smaller more graspable tasks.

Key Area	Information
Data security	Review of encryption usage, network access control, data security at rest, and transmissions.
Operation security	Review of security procedures, controls, and policies.
System security	Review of patching processes, hardening processes, role-based access, and privileged account management.
Network security	Review of network and security controls, security operation centre, antivirus configurations, and security monitoring capabilities.
Physical security	Review of disk encryption, biometric data, rolebased access controls, multifactor authentication, and other such measures.

Table 2.1: Key Areas of Cybersecurity Assessment (Jadhav, 2023, Table 1)

The security audit of *SCA Tool* and their product *SCA Tool Application* includes reviews across all key areas of table 2.1.

2.2 Information Security Standards

In the following two sections, two well known IS frameworks are presented. The *ISO/IEC 27000* standard family, which has become internationally established, and the *BSI IT-Grundschutz*, a fully comprehensive German framework.

2.2.1 ISO/IEC 27000 standard family

The *ISO/IEC 27000* standard family comprises an essential collection of international standards designed to facilitate effective ISM. Among these standards, *ISO/IEC 27001 (Information security management systems – Requirements)* is widely recognized as the world's foremost standard for ISMSs. This standard outlines the requirements for establishing, implementing, maintaining, and continually improving an ISMS within an organization.

Achieving *ISO/IEC 27001* certification can bring numerous advantages to organizations. It demonstrates a commitment to managing information securely and helps building trust with clients, stakeholders, and partners. Certified organizations often experience enhanced credibility, improved risk management, and a competitive edge in the marketplace. Moreover, the certification process itself encourages organizations to assess and improve their information security processes, fostering a culture of continuous improvement.

Within the *ISO/IEC 27000* standard family, a distinction is made between normative and informative standards. While *ISO/IEC 27001* as a normative standard contains requirements which must be fulfilled by the organization to be compliant with the norm, the *ISO/IEC 27002 (Information security controls)* is of informative character consisting of elucidations and recommendations on the implementation of the attachment *A* of the *ISO/IEC 27001*.

Brenner et al. (2024) illustrate, that the *ISO/IEC 27000* standard family encompasses more than 50 documents providing best practices for data protection and cyber resilience. Further normative standards are *ISO/IEC 27006 (Requirements for bodies providing audit and certification of information security management systems)*, *ISO/IEC 27009 (Sector-specific application of ISO/IEC 27001 – Requirements)*, and additionally *ISO/IEC 27701 (Extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management — Requirements and guidelines)*. Examples for additional informative standards are *ISO/IEC 27003 (Information security management systems – Guidance)*, *ISO/IEC 27004 (Information security management – Monitoring, measurement, analysis and evaluation)*, *ISO/IEC 27005 (Guidance on managing information security risks)*, and *ISO/IEC 27018 (Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors)*. While the mentioned informative standards are usable independently from the size of the organization or its sector, some standards are sector-specific, such as *ISO/IEC 27010 (Information security management for inter-sector and inter-organizational communications)* or *ISO/IEC 27017 (Code of practice for information security controls based on ISO/IEC 27002 for cloud services)*, or measure-specific, namely *ISO/IEC 27034 (Application security)* or *ISO/IEC 27035 (Information security incident management)*. Collectively, these standards enable organizations to effectively manage the security of a wide array of assets, including financial information, intellectual property, employee data, and data entrusted by third parties. Through the adoption of these standards, organizations can strengthen their overall information security framework and improve the protection of their vital assets against evolving cyber threats.

2.2.2 BSI IT-Grundschutz

Since 1994, the *BSI IT-Grundschutz* represents a foundational framework developed by the Federal Office for Information Security (BSI) in Germany, designed to enhance IS across various organizations. It covers all aspects that can be encountered in organizations. In addition to focusing on organization and personnel, it also encompasses Information Technology (IT) operations and production and manufacturing that utilize Industrial Control Systems (ICS) and components related to the Internet of Things (IoT). Central to the efforts of the BSI in promoting IS are several key standards, among which the *BSI Standard 200* series are particularly significant.

BSI Standard 200-1 establishes the framework for ISMSs, outlining the principles and requirements necessary for organizations to implement and maintain an effective ISMS. It provides a critical foundation for ensuring that the organization's IS practices are systematically integrated into its overall management structure.

This is complimented by *BSI Standard 200-2*, which outlines the *IT-Grundschutz* methodology. This standard offers a structured approach that organizations can adopt to analyze their security needs, identify potential vulnerabilities, and implement suitable protective measures. It serves as a guiding tool that helps organizations tailor their security measures according to their specific operational context and requirements.

To further refine the risk management process, *BSI Standard 200-3* introduces a comprehensive risk analysis based on *IT-Grundschutz*, providing organizations with a systematic method for assessing risks associated with their IT infrastructure. Through this analysis, organizations can identify, evaluate, and prioritize risks, enabling them to allocate resources effectively and address their most pressing security concerns.

In addition to these key standards, the BSI has developed further standards such as *BSI Standard 200-4*, which complements the management framework with Business Continuity Management (BCM).

The standards *BSI Standard 200-1* to *BSI Standard 200-4* collectively contribute to the development and the maintenance of a robust IS strategy, ensuring that organizations can fortify their defenses against potential threats (BSI, 2017a).

Schildt et al. (2023) describe the *IT-Grundschutz Compendium* as a central repository of best practices and guidelines drawn from these standards, so called *modules*, providing organizations with practical insights into implementing the *IT-Grundschutz*. The compendium details relevant measures that organizations can adopt to safeguard their information systems, facilitating a comprehensive approach to IS.

Moreover, the *IT-Grundschutz profiles* play a critical role in categorizing different information systems based on their specific security requirements. These *profiles* allow organizations to prioritize their security measures according to the criticality of their assets being protected, ensuring that resources are allocated efficiently where they are needed most. A *profile* of the own information system can be created. However, the BSI also makes *profiles*, that were already created by the community, available for download on its homepage[1]. Using an existing *profile*, which fits to the own organization due to similar security requirements, reduces the initial time and personnel expenses for the creation of a functional

[1] https://www.bsi.bund.de/dok/10027580

ISMS. They guide through the IS process tailored to these predefined security requirements.

As BSI (2017a) states, *BSI Standard 200-1* aligns perfectly with *ISO/IEC 27001*, while also incorporating the definitions from *ISO/IEC 27000* and the recommendations from *ISO/IEC 27002*. It offers a clear and structured approach that organizations can follow to fulfill ISMS requirements, regardless of the method they choose. Furthermore, *IT-Grundschutz*, building upon *BSI Standard 200-2*, clarifies the overall requirements and security measures from the previously mentioned ISO standards, providing users with valuable information, background context, and practical examples for implementation. The *IT-Grundschutz Compendium modules* detail the necessary actions to take, and the implementation recommendations deliver specific guidance on how to meet these requirements, including technical aspects. Therefore, utilizing *IT-Grundschutz* presents a reliable and effective way to meet the requirements of these ISO standards.

The single parts of the *IT-Grundschutz* such as the *BSI standards*, the *IT-Grundschutz Compendium* and the *IT-Grundschutz* profiles are distributed via the homepage of the BSI free of charge[2]. Additional resources such as a mapping table of the contents of the *IT-Grundschutz* to the ones of the *ISO/IEC 27001* and *ISO/IEC 27002*[3], online courses[4] and check lists[5] can be found there as well. Although some resources are available in English, the majority are exclusively offered in German.

Overall, the *BSI IT-Grundschutz* framework, along with its accompanying standards and resources, serves as a crucial roadmap for organizations aiming to enhance their IS posture. Through the systematic application of these methodologies, organizations can effectively manage risks and build resilience against emerging security threats, thereby fortifying their information assets and maintaining the trust of their stakeholders.

[2] https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/it-grundschutz_node.html

[3] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/IT-GS-Kompendium/Zuordnung_ISO_und_IT_Grundschutz_Edit_6.html

[4] https://www.bsi.bund.de/dok/10989992

[5] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/IT-GS-Kompendium/checklisten_2023.html?nn=128568

3 Vulnerability Assessment

In order to achieve a rapid improvement in the IS of *SCA Tool*, this Master's thesis identifies potential vulnerabilities and uses them to create policies and security mitigations. This approach prioritizes the areas with the most urgent need for action. In this chapter, possible vulnerabilities in *SCA Tool* and *SCA Tool Application* are exposed. At the beginning of each section, the commit hash of the *SCA Tool Application* source code is stated, that was affected by the vulnerability, or solely the date, when the issue was recorded, if the issue is not source code related. The specification of the date is important, as the vulnerability may have already been addressed by this time. Found security flaws may be false positives and not every found vulnerability is prone to be exploited due to IT infrastructural circumstances. However, a chain of security issues can lead to an exploitable serious security gap. That is why every layer should be secured as feasible as possible.

SCA Tool Application is running in a *Kubernetes*[1] cluster hosted in a Virtual Machine (VM), which is hosted in a Proxmox Virtual Environment (PVE)[2]. The `production` deployment is separated from the `testing` and `staging` deployments as it is running in its own VM. PVE and its setup are not part of this security audit. Solutions by *Cloudflare, Inc.*, which are placed in front of *SCA Tool Application*, and their configurations are excluded from this audit as well, but are tested slightly in context. Areas not mentioned in this thesis may be out of scope or already sufficiently addressed by the organization and therefore not explicitly stated.

During this Master's thesis, dynamic application security testing (DAST) was conducted against both the `testing` and `production` environments using Zed Attack Proxy (ZAP)[3] by Checkmarx. While `production` was not in scope, it was the only implementation delivered via Hypertext Transfer Protocol Secure (HTTPS) as stated in section 3.6. Some of the findings, that are addressed in the following sections, were identified by using other penetration testing tools as well. By

[1] https://kubernetes.io/

[2] https://www.proxmox.com/en/products/proxmox-virtual-environment/overview

[3] https://www.zaproxy.org/

regularly utilizing ZAP or similar tools for Dynamic Application Security Testing (DAST) as well as tool for Static Application Security Testing (SAST), vulnerabilities from multiple areas can be discovered at once, significantly simplifying security management. The ZAP reports can be found in appendices A (`testing`) and B (`production`).

3.1 Usage of Outdated and Vulnerable Libraries

*Affected commit **fecf785a2888e7f8f062975859c6d017185cfae3** of **2024-11-09**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.1**.*

A *CycloneDX Software Bill of Materials (SBOM)* was generated from the source code of *SCA Tool Application* by using the npm-package *@cyclonedx/cyclonedx-npm*[4]. It was installed and run by using the commands shown in listing 3.1 within the project folder of *SCA Tool Application*.

```
npm install --save-dev @cyclonedx/cyclonedx-npm;
npx cyclonedx-npm --output-file scatool.sbom
```

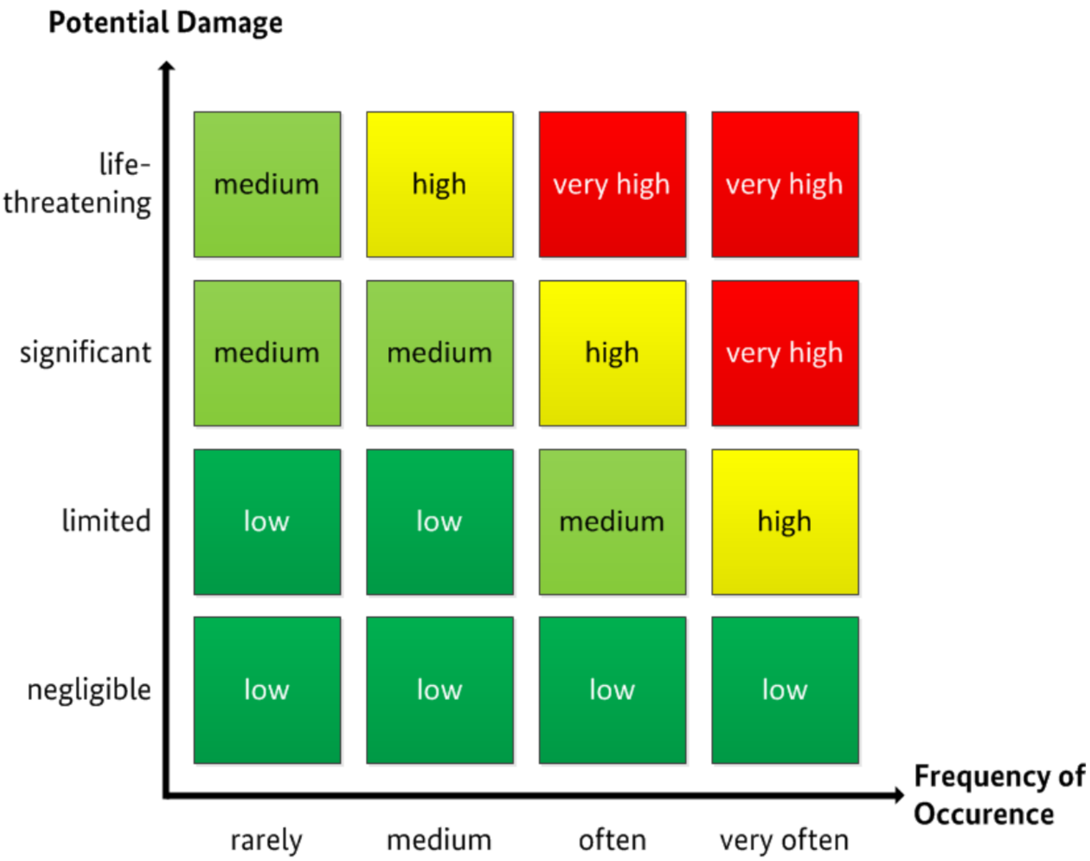
Listing 3.1: Creation of *CycloneDX SBOM* From *SCA Tool Application*

The generated SBOM `scatool.sbom` was then uploaded to a self-hosted instance of *OWASP Dependency-Track*[5] for analysis. The results are shown in figure 3.2. *Dependency-Track* lists twelve software components for *SCA Tool Application* as vulnerable. Their severity is sometimes shown as `Unassigned`, since *Dependency Track* checks in its default configuration the National Vulnerability Database (NVD) for recorded CVE, where no rating is provided yet and the vulnerability is still awaiting analysis. The yellow triangle with an exclamation mark indicates outdated components. The CVE-Identifier (ID) of the found vulnerable components were then cross-checked with cve.org[6], a public catalog of CVE records operated by *The MITRE Corporation*[7].

In the following, software components of *SCA Tool Application*, which are outdated and expose vulnerabilities, are listed regarding their score calculated with the newest Common Vulnerability Scoring System (CVSS) version. A score calculated with *CVSSv4.0* supersedes one calculated with *CVSSv3.1*, due to several improvements of *CVSSv4.0* over *CVSSv3.1* in the calculation process. It is to mention, that the score values the severity of the vulnerability and not the risk to the organization. The risk must be evaluated by the organization, since only they know their organizational context, their assets and their exposure to dam-

[4] https://www.npmjs.com/package/@cyclonedx/cyclonedx-npm
[5] https://dependencytrack.org/
[6] https://www.cve.org/
[7] https://www.mitre.org/

age, e.g., to penalty payments. BSI (2017b, Figure 3) illustrates a risk matrix in figure 3.1 to support the organization in their risk evaluation.



Potential Damage	rarely	medium	often	very often
life-threatening	medium	high	very high	very high
significant	medium	medium	high	very high
limited	low	low	medium	high
negligible	low	low	low	low

Figure 3.1: Risk Evaluation Matrix (BSI, 2017b, Figure 3)

The parameters `Potential Damage` and `Frequency of Occurence` [*sic*] must be defined, which can be different for every organization. This matrix is based on equation 3.1 and does not take the probability of exploitation of a vulnerability into account. Exemplary mappings from ambiguous delimitations to clear values are shown in the tables 3.1 and 3.2.

$$Risk = Potential\ Damage \times Frequency\ of\ Occurrence \tag{3.1}$$

Potential Damage	Monetary Damage
negligible	100 €
limited	1,000 €
significant	10,000 €
life-threatening	100,000 €

Table 3.1: Mapping of Potential Damage to Monetary Damage

Frequency of Occurrence	Temporal Frequency
rarely	every 5 years
medium	every year
often	every month
very often	every week

Table 3.2: Mapping of Frequency of Occurrence to Temporal Frequency

Based on the exemplary numbers provided in the tables 3.1 and 3.2, a potential significant damage, which could happen often, would result in a high risk to an enterprise as demonstrated in equation 3.2.

$$\mathit{Risk} = 10,000 \text{ €} \times \text{every month} \rightarrow \boxed{\text{high}} \tag{3.2}$$

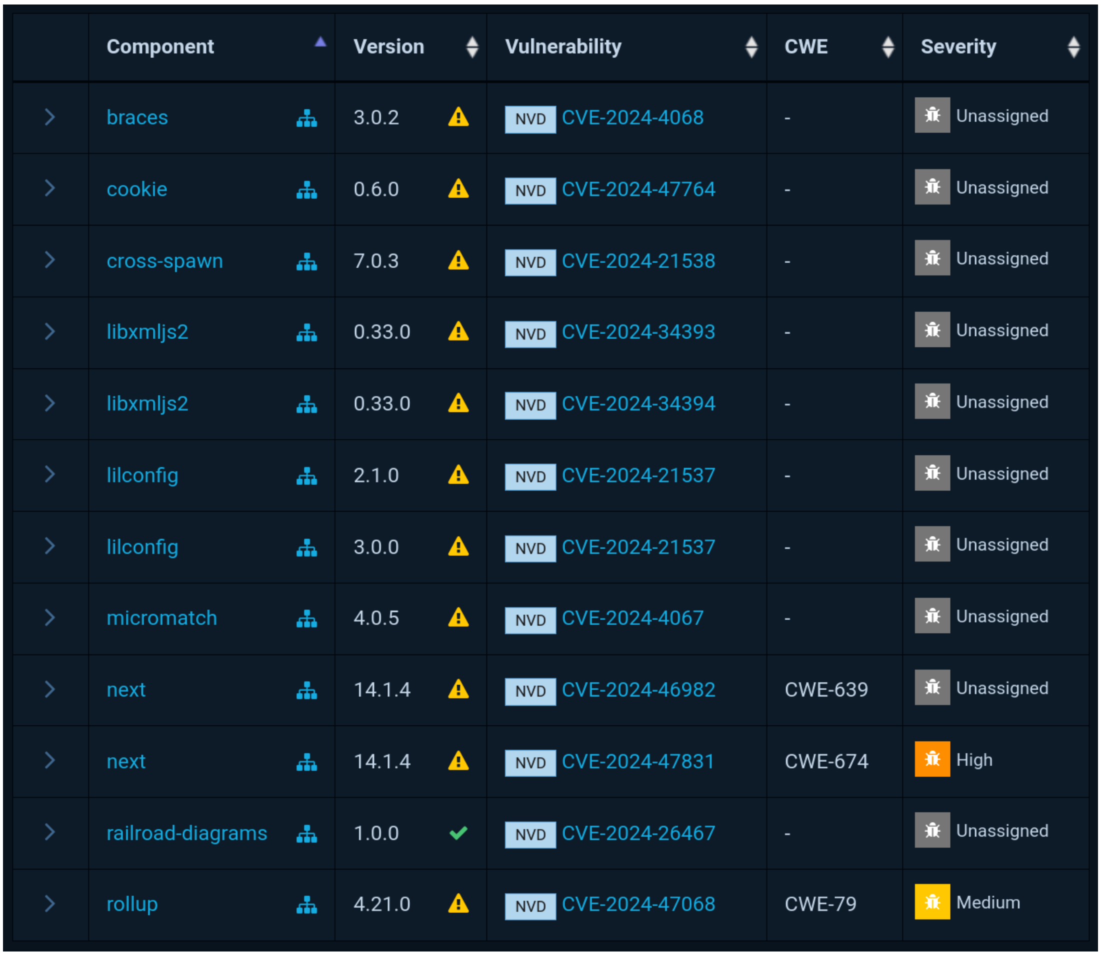
The tables 3.3 to 3.15 (without table 3.14) present the vulnerable and outdated libraries found with *Dependency-Track*. They consist of the following information: The clickable *CVE-ID*, which links to the corresponding record of cve.org. The *rating entity*, which can be a CVE Numbering Authority (CNA)[8] or an Authorized Data Publisher (ADP)[9]. A short description of the CVE record, which is copied from cve.org. The CVSS score, the resulting severity and the CVSS version, under which the score was calculated. The CVSS vector string shows the exact calculation of the score and links to the pre-filled CVSS score calculator at first.org[10].

A graphical and structured presentation of information is crucial in the field of security to enable prompt, informed decision-making and to engage non-expert personnel. By presenting data in a tabular format and pre-selecting information, such as if multiple CVSS scores are available, the CVE can be communicated effectively to the responsible parties.

[8] https://www.cve.org/ProgramOrganization/CNAs

[9] https://www.cve.org/ProgramOrganization/ADPs

[10] https://www.first.org/cvss/calculator/4.0



	Component	Version	Vulnerability	CWE	Severity
>	braces	3.0.2 ⚠	NVD CVE-2024-4068	-	Unassigned
>	cookie	0.6.0 ⚠	NVD CVE-2024-47764	-	Unassigned
>	cross-spawn	7.0.3 ⚠	NVD CVE-2024-21538	-	Unassigned
>	libxmljs2	0.33.0 ⚠	NVD CVE-2024-34393	-	Unassigned
>	libxmljs2	0.33.0 ⚠	NVD CVE-2024-34394	-	Unassigned
>	lilconfig	2.1.0 ⚠	NVD CVE-2024-21537	-	Unassigned
>	lilconfig	3.0.0 ⚠	NVD CVE-2024-21537	-	Unassigned
>	micromatch	4.0.5 ⚠	NVD CVE-2024-4067	-	Unassigned
>	next	14.1.4 ⚠	NVD CVE-2024-46982	CWE-639	Unassigned
>	next	14.1.4 ⚠	NVD CVE-2024-47831	CWE-674	High
>	railroad-diagrams	1.0.0 ✓	NVD CVE-2024-26467	-	Unassigned
>	rollup	4.21.0 ⚠	NVD CVE-2024-47068	CWE-79	Medium

Figure 3.2: CVEs Found With *OWASP Dependency-Track*

(HIGH - 8.8) lilconfig

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-21537</td><td><b>Rating Entity:</b> <i>Snyk</i></td></tr>
<tr><td colspan="3"><b>Description:</b>
Versions of the package lilconfig from 3.1.0 and before 3.1.1 are vulnerable to Arbitrary Code Execution due to the insecure usage of eval in the dynamicImport function. An attacker can exploit this vulnerability by passing a malicious input through the defaultLoaders function.</td></tr>
<tr><td><b>CVSS Score:</b> 8.8</td><td><b>Severity:</b> HIGH</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="3"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H/E:P</td></tr>
</table>

Table 3.3: Overview of *CVE-2024-21537*

Additional information: The npm-package *lilconfig* is used in two different versions throughout *SCA Tool Application*: 2.1.0 and 3.0.0.

(HIGH - 8.1) libxmljs2

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-34393</td><td colspan="2"><b>Rating Entity:</b> <i>JFrog</i></td></tr>
<tr><td colspan="4"><b>Description:</b>
libxmljs2 is vulnerable to a type confusion vulnerability when parsing a specially crafted XML while invoking a function on the result of attrs() that was called on a parsed node. This vulnerability might lead to denial of service (on both 32-bit systems and 64-bit systems), data leak, infinite loop and remote code execution (on 32-bit systems with the XML_PARSE_HUGE flag enabled).</td></tr>
<tr><td><b>CVSS Score:</b> 8.1</td><td colspan="2"><b>Severity:</b> HIGH</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="4"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H</td></tr>
</table>

Table 3.4: Overview of *CVE-2024-34393*

(HIGH - 8.1) libxmljs2

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-34394</td><td colspan="2"><b>Rating Entity:</b> <i>JFrog</i></td></tr>
<tr><td colspan="4"><b>Description:</b>
libxmljs2 is vulnerable to a type confusion vulnerability when parsing a specially crafted XML while invoking the namespaces() function (which invokes XmlNode::get_local_namespaces()) on a grand-child of a node that refers to an entity. This vulnerability can lead to denial of service and remote code execution.</td></tr>
<tr><td><b>CVSS Score:</b> 8.1</td><td colspan="2"><b>Severity:</b> HIGH</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="4"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H</td></tr>
</table>

Table 3.5: Overview of *CVE-2024-34394*

(HIGH - 7.5) braces

CVE-ID: CVE-2024-4068	**Rating Entity:** *Checkmarx*	
Description: The NPM package ‘braces‘, versions prior to 3.0.3, fails to limit the number of characters it can handle, which could lead to Memory Exhaustion. In ‘lib/parse.js‘, if a malicious user sends "imbalanced braces" as input, the parsing will enter a loop, which will cause the program to start allocating heap memory without freeing it at any moment of the loop. Eventually, the JavaScript heap limit is reached, and the program will crash.		
CVSS Score: 7.5	**Severity:** HIGH	**CVSS Version:** 3.1
CVSS Vector String: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H		

Table 3.6: Overview of *CVE-2024-4068*

(HIGH - 7.5) cross-spawn

CVE-ID: CVE-2024-21538	**Rating Entity:** *Snyk*	
Description: Versions of the package cross-spawn before 7.0.5 are vulnerable to Regular Expression Denial of Service (ReDoS) due to improper input sanitization. An attacker can increase the CPU usage and crash the program by crafting a very large and well crafted string.		
CVSS Score: 7.5	**Severity:** HIGH	**CVSS Version:** 3.1
CVSS Vector String: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H/E:P		

Table 3.7: Overview of *CVE-2024-21538*

(HIGH - 7.5) next

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-46982</td><td colspan="2"><b>Rating Entity:</b> <i>GitHub, Inc.</i></td></tr>
<tr><td colspan="4"><b>Description:</b>
Next.js is a React framework for building full-stack web applications. By sending a crafted HTTP request, it is possible to poison the cache of a non-dynamic server-side rendered route in the pages router (this does not affect the app router). When this crafted request is sent it could coerce Next.js to cache a route that is meant to not be cached and send a ‘Cache-Control: s-maxage=1, stale-while-revalidate‘ header which some upstream CDNs may cache as well. To be potentially affected all of the following must apply: 1. Next.js between 13.5.1 and 14.2.9, 2. Using pages router, & 3. Using non-dynamic server-side rendered routes e.g., ‘pages/dashboard.tsx‘ not ‘pages/blog/[slug].tsx‘. This vulnerability was resolved in Next.js v13.5.7, v14.2.10, and later. We recommend upgrading regardless of whether you can reproduce the issue or not. There are no official or recommended workarounds for this issue, we recommend that users patch to a safe version.</td></tr>
<tr><td><b>CVSS Score:</b> 7.5</td><td colspan="2"><b>Severity:</b> HIGH</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="4"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H</td></tr>
</table>

Table 3.8: Overview of *CVE-2024-46982*

(HIGH - 7.5) next

CVE-ID: CVE-2024-47831	**Rating Entity:** *NVD*	
Description: Next.js is a React Framework for the Web. Cersions on the 10.x, 11.x, 12.x, 13.x, and 14.x branches before version 14.2.7 contain a vulnerability in the image optimization feature which allows for a potential Denial of Service (DoS) condition which could lead to excessive CPU consumption. Neither the ‘next.config.js‘ file that is configured with ‘images.unoptimized‘ set to ‘true‘ or ‘images.loader‘ set to a non-default value nor the Next.js application that is hosted on Vercel are affected. This issue was fully patched in Next.js ‘14.2.7‘. As a workaround, ensure that the ‘next.config.js‘ file has either ‘images.unoptimized‘, ‘images.loader‘ or ‘images.loaderFile‘ assigned.		
CVSS Score: 7.5	**Severity:** HIGH	**CVSS Version:** 3.1
CVSS Vector String: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H		

Table 3.9: Overview of *CVE-2024-47831*

Two different scores were given for this CVE. *GitHub, Inc.* scored it as 5.9, medium severity, with a high *Attack Complexity*, which is still shown at cve.org. This was later recalculated by *NVD* to 7.5, high, with a low *Attack Complexity* and therefore a higher *Exploitability Score*. Accordingly, the higher value is indicated in 3.9, even though cve.org displays the lower value.

(MEDIUM - 6.9) cookie

CVE-ID: CVE-2024-47764	**Rating Entity:** *GitHub, Inc.*	
Description: cookie is a basic HTTP cookie parser and serializer for HTTP servers. The cookie name could be used to set other fields of the cookie, resulting in an unexpected cookie value. A similar escape can be used for path and domain, which could be abused to alter other fields of the cookie. Upgrade to 0.7.0, which updates the validation for name, path, and domain.		
CVSS Score: 6.9	**Severity:** MEDIUM	**CVSS Version:** 4.0
CVSS Vector String: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N		

Table 3.10: Overview of *CVE-2024-47764*

(MEDIUM - 6.1) railroad-diagrams

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-26467</td><td colspan="2"><b>Rating Entity:</b> <i>CISA-ADP</i></td></tr>
<tr><td colspan="4"><b>Description:</b>
A DOM based cross-site scripting (XSS) vulnerability in the component generator.html of tabatkins/railroad-diagrams before commit ea9a123 allows attackers to execute arbitrary Javascript via sending a crafted URL.</td></tr>
<tr><td><b>CVSS Score:</b> 6.1</td><td colspan="2"><b>Severity:</b> MEDIUM</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="4"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N</td></tr>
</table>

Table 3.11: Overview of *CVE-2024-26467*

(MEDIUM - 6.1) rollup

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-47068</td><td colspan="2"><b>Rating Entity:</b> <i>GitHub, Inc.</i></td></tr>
<tr><td colspan="4"><b>Description:</b>
Rollup is a module bundler for JavaScript. Versions prior to 2.79.2, 3.29.5, and 4.22.4 are susceptible to a DOM Clobbering vulnerability when bundling scripts with properties from ‘import.meta‘ (e.g., ‘import.meta.url‘) in ‘cjs‘/‘umd‘/‘iife‘ format. The DOM Clobbering gadget can lead to cross-site scripting (XSS) in web pages where scriptless attacker-controlled HTML elements (e.g., an ‘img‘ tag with an unsanitized ‘name‘ attribute) are present. Versions 2.79.2, 3.29.5, and 4.22.4 contain a patch for the vulnerability.</td></tr>
<tr><td><b>CVSS Score:</b> 6.1</td><td colspan="2"><b>Severity:</b> MEDIUM</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="4"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N</td></tr>
</table>

Table 3.12: Overview of *CVE-2024-47068*

(MEDIUM - 5.3) micromatch

<table>
<tr><td colspan="2"><b>CVE-ID:</b> CVE-2024-4067</td><td><b>Rating Entity:</b> <i>Checkmarx</i></td></tr>
<tr><td colspan="3"><b>Description:</b>
The NPM package ‘micromatch‘ prior to 4.0.8 is vulnerable to Regular Expression Denial of Service (ReDoS). The vulnerability occurs in ‘micromatch.braces()‘ in ‘index.js‘ because the pattern ‘.*‘ will greedily match anything. By passing a malicious payload, the pattern matching will keep backtracking to the input while it doesn’t find the closing bracket. As the input size increases, the consumption time will also increase until it causes the application to hang or slow down. There was a merged fix but further testing shows the issue persists. This issue should be mitigated by using a safe pattern that won’t start backtracking the regular expression due to greedy matching. This issue was fixed in version 4.0.8.</td></tr>
<tr><td><b>CVSS Score:</b> 5.3</td><td><b>Severity:</b> MEDIUM</td><td><b>CVSS Version:</b> 3.1</td></tr>
<tr><td colspan="3"><b>CVSS Vector String:</b>
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L</td></tr>
</table>

Table 3.13: Overview of *CVE-2024-4067*

The original advisory by *Checkmarx* AppSec Analyst Mário Teixeira stated a CVSS score of 7.5, high severity (Teixeira, 2024), which was also initially reflected in the NVD in May 2024. However, about one week later, *Checkmarx* reported an update to the CVSS vector, which is displayed by cve.org and NVD. The new score is 5.3, medium severity. Comparing the deprecated to the updated CVSS vector as shown in table 3.14 leads to the reason behind the update: The CVSS score was calculated incorrectly.

Deprecated:	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H
Updated:	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L

Table 3.14: Comparison of CVSS Vectors Reported by *Checkmarx*

The initially reported description of the vulnerability shows, that the effect on availability is low, as it "can cause the application to hang or slow down" (Teixeira, 2024). Whereas a high impact on availability means a total loss of availability, be it through one-off or repeated exploitation of the vulnerability. This example shows the importance for the risk assessment to consider not only the CVSS score but also the calculation process.

(N/A) nanoid

CVE-ID: CVE-2024-55565	**Rating Entity:** *CISA-ADP*	
Description: nanoid (aka Nano ID) before 5.0.9 mishandles non-integer values. 3.3.8 is also a fixed version.		
CVSS Score: 4.3	**Severity:** medium	**CVSS Version:** 3.1
CVSS Vector String: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:L/A:N		

Table 3.15: Overview of *CVE-2024-55565*

Conducting software checks for CVEs can often be a time-consuming process. Furthermore, if these checks are overlooked, managing updates across multiple software versions may lead to significant challenges. As a result, implementing regular automated checks is essential to ensure timely identification and remediation of vulnerabilities. Section 5.1 gives instructions on how to automate the checking process and speed up the decision making of updating vulnerable or outdated libraries.

3.2 Misconfigured Identity and Access Management

Affected commit ***e33cc7f692ce55e56e751c192642bdf601a6730f*** *of* ***2024-12-28****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.2****.*

The Identity and Access Management (IAM)-solution *Ory Kratos*[11] is in use. By relying on an established solution instead of writing their own, smaller enterprises in particular can ensure that the IS protection goals of confidentiality, integrity and availability are met. Nevertheless, the solution must be configured accordingly. The *Ory Kratos* configuration used in *SCA Tool Application* is not configured properly and therefore not production-ready.

3.2.1 Insecure Password Policy

The mitigation strategy to this finding is detailed in (sub)section ***5.2.1****.*

The password policy shown in listing 3.2 is insecure. The current configuration does not check the user's password against a blocklist as recommended by *NIST*

[11] https://www.ory.sh/kratos/

(Temoshok, 2024, Chapter 3.1.1.2.). *Ory Kratos* allows for using the API of *Have I Been Pwned (HIBP)*[12] or a selfhosted blocklist.

```
selfservice:
...
  methods:
    password:
      enabled: true
      config:
        haveibeenpwned_enabled: false
```

Listing 3.2: *kratos.yaml*: Insecure Password Policy Values

3.2.2 Missing Multi-Factor Authentication

*The mitigation strategy to this finding is detailed in (sub)section **5.2.2**.*

There is no Multi-Factor Authentication (MFA) set up for *SCA Tool Application*. If an attacker gains possession of the user's username and password, there are no security measures in place to prevent unauthorized login, data exfiltration, or modifications to sensitive features.

3.2.3 Logging Sensitive Values

*The mitigation strategy to this finding is detailed in (sub)section **5.2.3**.*

By setting `leak_sensitive_values: true` in the configuration of the `testing` deployment as displayed in listing 3.3, sensitive data such as PII, tokens, etc. are logged. This is not the case for the configurations of `staging` and `production`. In a local development environment, setting the value to `true` is useful for troubleshooting, but should not be set by default in a shared configuration file. There is the risk that, due to human error, this configuration will be used in a production environment. By an unintentional leak of log files to unauthorized third parties or through access by employees of the organization, who have access to central log analysis tools, such as a Security Information and Event Management (SIEM)-tool, the protection goal of confidentiality could be violated. Even without a deployment of this configuration to `production`, it may be the case, that developers mistakenly try to log into the `testing` version of *SCA Tool Application* with their `production` credentials, which then would be logged.

[12] https://haveibeenpwned.com/API/v3

```yaml
log:
   level: debug
   format: text
   leak_sensitive_values: true
```

Listing 3.3: *kratos.yaml*: Logging of Sensitive Values

3.2.4 Insecure Password Hashing

*The mitigation strategy to this finding is detailed in (sub)section **5.2.4**.*

As displayed in listing 3.4, *bcrypt* is used in the `testing` deployment for password hashing with a `cost` parameter of `8`, which is too low for modern Central Processing Units (CPUs) used for cracking the hash. Given that a secure `cost` factor of `12` is applied to `staging` and `production`, the presence of a lower cost factor is not critical. However, there is a risk that developers may inadvertently transfer the `cost` parameter from `testing` to `staging` or even `production`.

Additionally, issues related to loading times may not be identified during the development phase due to the less time-intensive `cost` factor. Hashing with the secure `cost` factor of `12` requires a processing time that is 2^4 times longer than hashing with the insecure `cost` factor of `8`. Nevertheless, the processing time would remain under 0.5 seconds per hash, allowing login masks to still feel smooth to the user (Düsterhus, 2023). As Thompson (2016) points out, the parameter must be increased roughly every 18 months, which puts administrative overhead on the developers if no automation is used to change it automatically.

bcrypt itself, while still being secure with a high enough `cost` factor, is no longer recommended by *Open Worldwide Application Security Project (OWASP)* and should only be used for legacy systems (OWASP, 2024).

```yaml
hashers:
   bcrypt:
      cost: 8
```

Listing 3.4: *kratos.yaml*: Insecure `cost` Factor

3.2.5 Missing Re-Authentication for Sensitive Features

*Affected commit **5104bab58f4a72980ef754b655c2e5cb66eabb82** of **2025-02-20**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.2.5**.*

When the user updates their profile or password in the `User Settings`, they are not asked to verify themselves by providing their (old) password within the first 15 minutes after login. Should a user leave their end device unattended and unlocked or forget to log themselves out of a shared computer, an attacker can change sensitive values within this time frame and gain control of the user's account. This also applies if an attacker has successfully stolen the user's session token.

3.2.6 Multiple Active User Sessions

Affected commit ***5104bab58f4a72980ef754b655c2e5cb66eabb82*** *of* ***2025-02-20****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.2.6****.*

SCA Tool Application permits multiple active user sessions for an individual user. If an attacker were to gain access to the user's device, for example through theft, or *session hijacking*[13] by acquiring the user's `ory_kratos_session`-cookie, they would have unrestricted access to the user's account without the user's awareness.

The corresponding `flow` in *kratos.yaml* is missing the `revoke_active_sessions`-hook.

3.3 Reused Plain Text Credentials in Source Code

Affected commit ***e33cc7f692ce55e56e751c192642bdf601a6730f*** *of* ***2024-12-28****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.3****.*

The source code of *SCA Tool* contains non-encrypted credentials, which are shown in listings 3.5 to 3.12. The credentials are stored in data-oriented files of type YAML Ain't Markup Language (YAML) at `/scatool/k8s/charts/secrets/`. These files are uploaded to github.com, a cloud-based platform for collaborative working on source code. A leak of the source code either due to incorrect configuration of the online repository or due to an security issue on part of *GitHub, Inc.* also results in a leak of the credentials. For cryptographic tasks, such as encrypting, decrypting, generating and validating signatures, *Ory Kratos* uses secrets with a high entropy of more than 256 bit, which is rendered useless if the secrets are leaked. As secrets are reused across multiple deployments, a leak directly affects several stages.

Furthermore, the plain text credentials can be observed by an attacker performing *shoulder surfing*[14]. The attacker does not necessarily need to be an advanced adversary. In times of social media platforms, such as TikTok[15] or Instagram[16], it is not uncommon for users of these platforms to use their smartphones to create

[13] https://capec.mitre.org/data/definitions/593.html
[14] https://capec.mitre.org/data/definitions/508.html
[15] https://www.tiktok.com/en/
[16] https://www.instagram.com/

short films or photos of other people and their lives and then upload this content to the social media platforms without respecting their privacy. This spreads the captured sensitive information on the internet and therefore ends up in the hands of criminals. Especially student developers, who do not have their own office, but instead develop in the library or on the train, are prone to be victims of this attack type.

In addition to that, especially inexperienced developers are inadvertently trained to use passwords in plain text within the source code.

```
data:
   dsn: {{ "postgres://REDACTED:REDACTED@REDACTED:REDACTED/app?
      ↪ sslmode=require&max_conns=20&max_idle_conns=4" | b64enc
      ↪ | quote }}
   secretsDefault: {{ "REDACTED" | b64enc | quote }}
   secretsCookie: {{ "REDACTED" | b64enc | quote }}
   secretsCipher: {{ "REDACTED" | b64enc | quote }}
   smtpConnectionURI: {{
      ↪ "smtps://REDACTED:REDACTED@REDACTED:REDACTED/
      ↪ ?skip_ssl_verify=true" | b64enc | quote }}
```

Listing 3.5: *kratos.yaml*: Plain Text Credentials

```
stringData:
   CRYPTO_KEY_VALUE: "REDACTED"
```

Listing 3.6: *longhorn.yaml*: Plain Text Credentials

```
stringData:
   config.env: |-
   export MINIO_ROOT_USER="REDACTED"
   export MINIO_ROOT_PASSWORD="REDACTED"
```

Listing 3.7: *minio.yaml*: Plain Text Credentials

```
data:
   username: {{ "REDACTED" | b64enc | quote }}
   password: {{ "REDACTED" | b64enc | quote }}
...
data:
   username: {{ "REDACTED" | b64enc | quote }}
   password: {{ "REDACTED" | b64enc | quote }}
```

Listing 3.8: *postgres-main-cluster.yaml*: Plain Text Credentials

```
data:
  username: {{ "REDACTED" | b64enc | quote }}
  password: {{ "REDACTED" | b64enc | quote }}
...
data:
  username: {{ "REDACTED" | b64enc | quote }}
  password: {{ "REDACTED" | b64enc | quote }}
```

Listing 3.9: *postgres-ory-cluster.yaml*: Plain Text Credentials

```
data:
  ...
  password: {{ "REDACTED" | b64enc | quote }}
  ...
  username: {{ "REDACTED" | b64enc | quote }}
  default_user.conf: {{ "default_user = REDACTED\ndefault_pass
    ↪ = REDACTED" | b64enc | quote }}
```

Listing 3.10: *rabbitmq-cluster.yaml*: Plain Text Credentials

```
dsn: "postgres://REDACTED:REDACTED@REDACTED:REDACTED/app
  ↪ ?sslmode=require&max_conns=20&max_idle_conns=4"
...
courier:
  smtp:
    connection_uri:
      ↪ smtps://REDACTED:REDACTED@REDACTED:REDACTED/
      ↪ ?skip_ssl_verify=true
```

Listing 3.11: *kratos.yaml*: Plain Text Credentials

```
monolithProperties: |
    ...
    spring.datasource.username=REDACTED
    spring.datasource.password=REDACTED
    ...
    spring.rabbitmq.username=REDACTED
    spring.rabbitmq.password=REDACTED
    ...
    minio.accessKey=REDACTED
    minio.secretKey=REDACTED
    ...

workerProperties: |
    ...
    spring.rabbitmq.username=REDACTED
    spring.rabbitmq.password=REDACTED
    ...
    minio.accessKey=REDACTED
    minio.secretKey=REDACTED
```

Listing 3.12: *values.yaml*: Plain Text Credentials

3.4 Use of Default Credentials

Affected commit ***e33cc7f692ce55e56e751c192642bdf601a6730f*** *of* ***2024-12-28****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.4****.*

Some credentials of section 3.3 are not only in plain text but are also default credentials of the services in use and utilized across multiple stages.

Generally, default credentials are published in software, its documentation, in forums, in credentials lists and more. This finding puts *SCA Tool Application* at risk, since an attacker can successfully carry out an attack and exfiltrate or change information in a very short time with fewer resources.

Currently, the monitoring solutions are not set up to identify the use of default credentials. As these credentials are valid, their utilization does not trigger any alerts. Furthermore, because default credentials are more likely to be tried by an adversary if they know the targeted system, the threshold for incorrect login attempts is unlikely to be reached, resulting in the absence of alerts as well. Consequently, the utilization of default credentials enables an attacker to operate undetected and navigate through the system without raising any alarms.

3.5 Insecure Implementation of API Key Generation

The mitigation strategy to this finding is detailed in (sub)section ***5.5****.*

Concurrently with the analysis of the API key generation implementation, a new version was developed, allowing for an analysis of both implementations.

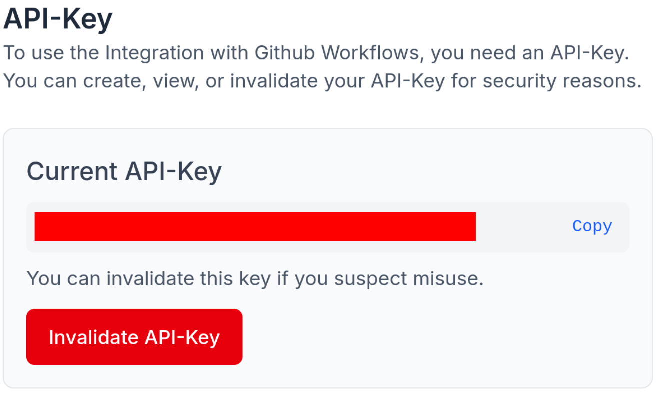
3.5.1 First Implementation

Affected commit ***1a003870032b02ef92144cdac6151ef1329fb9f*** *of* ***2025-02-15****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.5.1****.*

The user can create an API key to use *SCA Tool Application* in combination with *GitHub Actions workflows*. This API key can be set at `User Settings >` ↪ `Organization`.

After the button `Create API-Key` is clicked, the API key is generated on the server and displayed in plain text to the user. The `Copy`-link does not work and throws the error message `Failed to copy API-Key`.

After reloading the webpage, the API key is still shown in plain text. This poses a risk, as the API key can be stolen by means of *shoulder surfing*, for example. Attackers who have short access to the user's account can also retrieve the API key and use it to access *SCA Tool Application* unnoticed over a long period of time. The API key does not expire automatically, which puts the user to risk if the API key gets leaked unknowingly to the user.



API-Key

To use the Integration with Github Workflows, you need an API-Key.
You can create, view, or invalidate your API-Key for security reasons.

Current API-Key

[redacted] Copy

You can invalidate this key if you suspect misuse.

Invalidate API-Key

Figure 3.3: API Key Displayed in Plain Text

If the `Invalidate API-Key`-button is clicked, the API key is invalidated. This is done by means of a `DELETE`-request to the server of *SCA Tool Application*. The API key is sent directly in the path of the Uniform Resource Locator (URL) as shown in listing 3.13. This implementation is not secure. The URL path and header are encrypted in transit using Transport Layer Security (TLS) and are therefore not visible at first to third parties. However, the URL paths and headers are sometimes logged at stages where the encryption terminates. Cloudflare, Inc. (2021) also writes in their whitepaper about data privacy, that they log some Hypertext Transfer Protocol (HTTP) request metadata, such as the HTTP headers. If the delete request does not reach the server of *SCA Tool Application* or there is a bug in the delete instruction on server-side, this API key remains valid and has been recorded in plain text in several places.

```
DELETE /api/web/api-key/a2a6515e-79dc-4184-a3c4-317fce7899ae
    ↪ HTTP/1.1
```

Listing 3.13: Syllabus of HTTP Request Header

Furthermore, if the user clicks the `Create API-Key`-button and then the `Invalidate ↪ API-Key`-button in quick succession, another API key is displayed after the third time the invalidating button was pushed. If the user then clicks the `Invalidate ↪ API-Key`-button again, a new API key is shown. The user has to press the invalidating button at least three times in a row to get back to the original mask. The state of the database and which key is valid, could not be checked.

3.5.2 Second Implementation

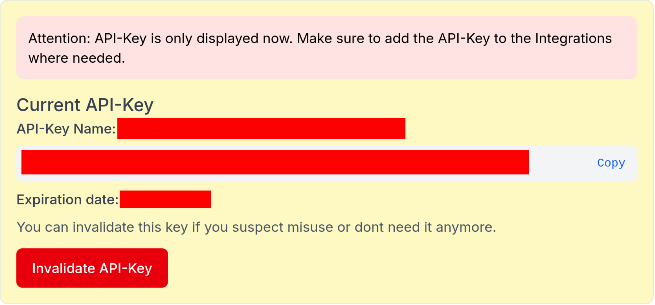
*Affected commit **fa53cb05429ff438e2433814149996ac3592b2459** of **2025-02-27**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.5.2**.*

After the first examination of the API key generation implementation, the developer reached out for assistance on the advice of the supervisor. During the meeting, it became clear that a new version of the API was already being worked on, which included some security-relevant features. The new version reveals the API key to the user only once, directly after creation. An expiration date from `1 Month` up to `No Expiration` can be set via a dropdown menu. For better usability, a name can be given to the API key. The API key as shown in figure 3.4 now consists of a prefix based on the expiration date and a random 32-bit string.

While the new implementation is considered more secure than the old one, there are still some issues to mitigate. First, the field of the API key name is not limited in number of characters, neither client-side nor server-side. An adversary can store large text blocks in the Database (DB), which can exhaust resources of the DB server, such as disk space, memory consumption and CPU power. If an adversary could misuse this field to inject arbitrary code, was not tested.

There is no benefit of showing the API key in plain text to the user. Due to its length and random character, the user will most likely copy the string instead of writing it down or remembering it.

The API key includes a prefix, which is used for DB-functions, such as finding keys by expiration date, and is meant to add security, since the attacker does not know, if the full string is stored hashed in the DB or only the last 32-bit string. However, since the prefix always has the same structure, the information could be stored in a separate column.



Attention: API-Key is only displayed now. Make sure to add the API-Key to the Integrations where needed.

Current API-Key

API-Key Name: [redacted]

[redacted] Copy

Expiration date: [redacted]

You can invalidate this key if you suspect misuse or dont need it anymore.

Invalidate API-Key

Figure 3.4: New API Key Generation Implementation

The prefix does not add real value to the security of the API key. If the DB gets leaked, *Rainbow Table Password Cracking*[17] against the DB can be performed. If an adversary created an API key before the DB was leaked, determining the hashing process is trivial and the cracking procedure will cost the same as without the prefix. Additionally if the key is leaked in plaintext, an adversary will know just by the key itself, which platform's API they can misuse and for how long. Therefore, as Smith (2022) concluded, security by obscurity may advert non-professional adversaries and increase costs on the attacker's side, but will not hinder a dedicated and knowledgeable attacker.

3.6 Non-Production Deployments without HTTPS

Affected commit ***e33cc7f692ce55e56e751c192642bdf601a6730f*** *of* ***2024-12-28****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.6****.*

Casola et al. (2024) express, that the `testing` environment should be as close as possible to the `production` environment. This allows comprehensive security testing without financial impact on the production side.

The `testing` and `staging` deployments of *SCA Tool Application* are running on a different VM than the `production` version. However, the `production` version running behind solutions by *Cloudflare, Inc.*, has a nameserver entry and is carried

[17] https://capec.mitre.org/data/definitions/55.html

out via HTTPS. The `testing` and `staging` deployments are only accessible by Internet Protocol (IP) address and via HTTP despite *Traefik* is shipping its own SSL/TLS-certificate.

Credentials associated with test accounts are transmitted in an unencrypted manner from the client to *SCA Tool Application*. Consequently, should a developer inadvertently log in using their `production` credentials on the `staging` version of the application or vice versa, a malicious actor may intercept these credentials during transmission. This vulnerability extends to all security tests conducted against the application, thereby enabling an Adversary-in-the-Middle (AiTM) to gain insights into the testing methodologies employed and the potential attack vectors that may be effective. Furthermore, essential security features, such as secure cookie implementation, cannot be tested under these conditions.

3.7 Missing and Insecure HTTP Headers

Affected commit ***fa53cb05429ff438e2433814l4996ac3592b2459*** *of* ***2025-02-27****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.7****.*

SCA Tool Application is not delivered with secure HTTP headers. Running the security-oriented HTTP headers analyzer *humble*[18] as shown in listing 3.14 against the `testing` deployment at `http://REDACTED` leads to the results displayed in listing 3.15.

```
humble -u http://REDACTED -b
```

Listing 3.14: *humble* Command

[18] https://github.com/rfc-st/humble

```
...

[2. Missing HTTP Security Headers]

Cache-Control
Clear-Site-Data
Cross-Origin-Embedder-Policy
Cross-Origin-Opener-Policy
Cross-Origin-Resource-Policy
Content-Security-Policy
(*) NEL
(*) Permissions-Policy
Referrer-Policy
Strict-Transport-Security
X-Content-Type-Options
X-Permitted-Cross-Domain-Policies
X-Frame-Options

...

[4. Deprecated HTTP Response Headers/Protocols and Insecure Values]

Content-Type (Unsafe Value)
Etag (Potentially Unsafe Header)
HTTP (URL Via Unsafe Scheme)
```

Listing 3.15: *humble* Results

The `production` version at https://app.scatool.com was not part of this security audit. However, it was checked for reference purposes if the *Cloudflare* configuration is set up to ship the missing HTTP headers. This is not the case.

The full reports are appended as appendix C (`testing`) and D (`production`).

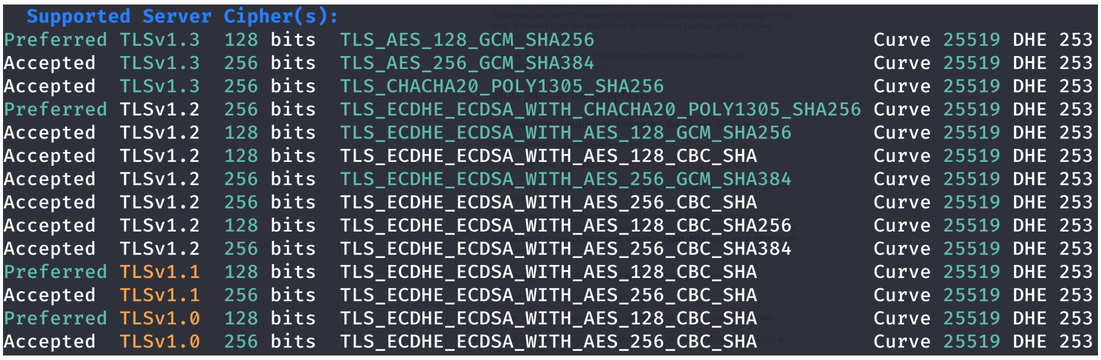
3.8 Insecure Protocols and Ciphers

Affected commit ***5d7e7ad723efb7b165c75d1f5e824a6cad0d5715*** *of* ***2025-02-26****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.8****.*

The deployment of the production version of *SCA Tool Application* supports insecure protocols and cipher suites. This can be checked with the command shown in listing 3.16. Figure 3.5 displays part of the output, which discloses the use of the protocols *TLSv1.0* and *TLSv1.1*. Moriarty and Farrell (2021) state, that these versions are rendered insecure and must not be used.

```
sslscan --iana-names app.scatool.com
```

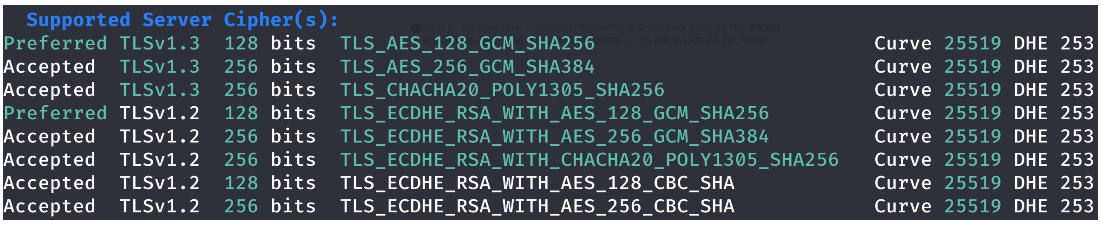
Listing 3.16: Command `sslscan` against `production`



```
 Supported Server Cipher(s):
Preferred TLSv1.3  128 bits  TLS_AES_128_GCM_SHA256                        Curve 25519 DHE 253
Accepted  TLSv1.3  256 bits  TLS_AES_256_GCM_SHA384                        Curve 25519 DHE 253
Accepted  TLSv1.3  256 bits  TLS_CHACHA20_POLY1305_SHA256                  Curve 25519 DHE 253
Preferred TLSv1.2  256 bits  TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 Curve 25519 DHE 253
Accepted  TLSv1.2  128 bits  TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256       Curve 25519 DHE 253
Accepted  TLSv1.2  128 bits  TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA          Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384       Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA          Curve 25519 DHE 253
Accepted  TLSv1.2  128 bits  TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256       Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384       Curve 25519 DHE 253
Preferred TLSv1.1  128 bits  TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA          Curve 25519 DHE 253
Accepted  TLSv1.1  256 bits  TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA          Curve 25519 DHE 253
Preferred TLSv1.0  128 bits  TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA          Curve 25519 DHE 253
Accepted  TLSv1.0  256 bits  TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA          Curve 25519 DHE 253
```

Figure 3.5: Output of `sslscan` Against `production`

As stated in section 3.6, the non-production deployments are not enforcing HTTPS, but are offering it. As seen in figure 3.6, the output of both deployments at `REDACTED` (`testing`) and `REDACTED` (`staging`) show the solely use of *TLSv1.2* and *TLSv1.3*, which is recommended.



```
 Supported Server Cipher(s):
Preferred TLSv1.3  128 bits  TLS_AES_128_GCM_SHA256                        Curve 25519 DHE 253
Accepted  TLSv1.3  256 bits  TLS_AES_256_GCM_SHA384                        Curve 25519 DHE 253
Accepted  TLSv1.3  256 bits  TLS_CHACHA20_POLY1305_SHA256                  Curve 25519 DHE 253
Preferred TLSv1.2  128 bits  TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256         Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384         Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256   Curve 25519 DHE 253
Accepted  TLSv1.2  128 bits  TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA            Curve 25519 DHE 253
Accepted  TLSv1.2  256 bits  TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA            Curve 25519 DHE 253
```

Figure 3.6: Output of `sslscan` Against `testing` and `staging`

However, Al Fardan and Paterson (2013) discovered vulnerabilities in TLS and Datagram Transport Layer Security (DTLS) that enable an AiTM attacker to retrieve plaintext from a TLS/DTLS connection when using Cipher Block Chaining (CBC)-mode encryption. Stevens et al. (2017) proved the existence of successful collision attacks against Secure Hash Algorithm 1 (SHA-1), which was already deprecated by National Institute of Standards and Technology (NIST) in 2011. Therefore, following ciphers are seen as weak due to their use of CBC and SHA-1 and must not be used:

- `TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA`
- `TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA`

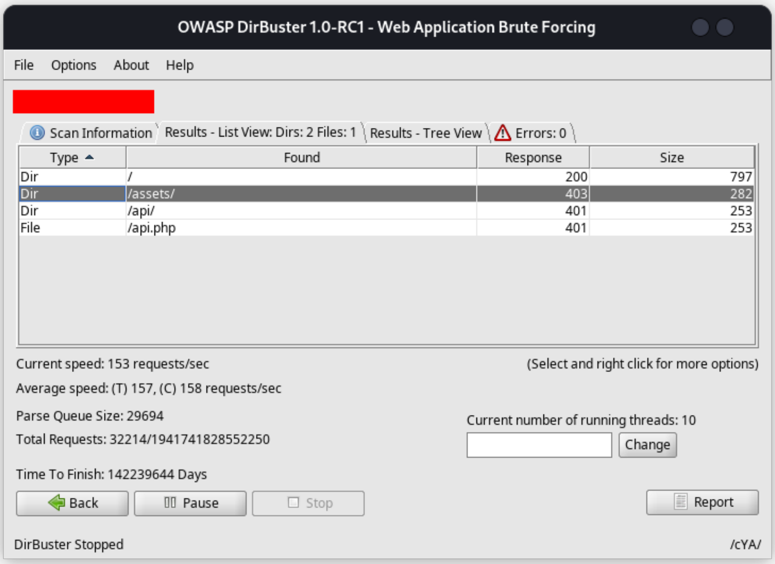
Section 5.8 describes ways on how to find secure ciphers to use for HTTPS.

3.9 Published Information About Infrastructure

Affected commit ***07d45e276b7c8cffaf3cd77ea9dffdcfef619760*** *of* ***2025-02-06****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.9****.*

Information about the infrastructure of *SCA Tool Application* is disclosed to the public. By running a directory traversal attack with *OWASP DirBuster*[19] against the `testing` deployment of *SCA Tool Application* at `http://REDACTED`, the response `403 Forbidden` gets received at `http://REDACTED/assets/` as shown in figure 3.7. In contrast to other error pages of *SCA Tool Application*, such as the one for `404 ↪ Not Found`, this error message is not a custom page. Instead it is the default page of an *nginx* web server revealing its version in the response's header and body as shown in figure 3.8. Vulnerable versions of *nginx* are listed on their homepage[20].

Revealing information about the underlying technology stack can make the reconnaissance easier for adversaries. It allows them to tailor their attacks to the victim's infrastructure and remain possibly unobserved during the execution. While the disclosure of the latest version number of one of the most popular web servers does not necessarily represent a risk, the publication of an older version number may directly expose a vulnerability to the attacker.


OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing
File
Options
About
Help
Scan Information
Results - List View: Dirs: 2 Files: 1
Results - Tree View
Errors: 0
Back
Pause
Stop
Change
Report

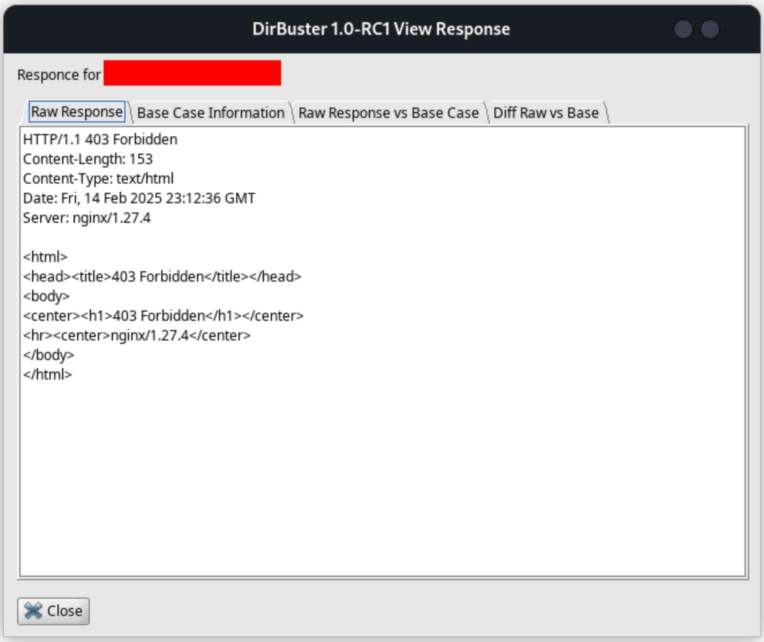
[redacted]

Type	Found	Response	Size
Dir	/	200	797
Dir	/assets/	403	282
Dir	/api/	401	253
File	/api.php	401	253

Current speed: 153 requests/sec
(Select and right click for more options)
Average speed: (T) 157, (C) 158 requests/sec
Parse Queue Size: 29694
Current number of running threads: 10
Total Requests: 32214/1941741828552250
Time To Finish: 142239644 Days
DirBuster Stopped
/cYA/

Figure 3.7: *OWASP DirBuster*: Directory Traversal Attack

[19] https://wiki.owasp.org/index.php/Category:OWASP_DirBuster_Project
[20] https://nginx.org/en/security_advisories.html



Responce for [redacted]

```
HTTP/1.1 403 Forbidden
Content-Length: 153
Content-Type: text/html
Date: Fri, 14 Feb 2025 23:12:36 GMT
Server: nginx/1.27.4

<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx/1.27.4</center>
</body>
</html>
```

Figure 3.8: *OWASP DirBuster*: *nginx* 403 Error Page

3.10 Missing Rate Limiting

Affected commit ***07d45e276b7c8cffaf3cd77ea9dffdcfef619760*** *of* ***2025-02-06****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.10****.*

The `production` deployment of *SCA Tool Application* is running behind solutions by *Cloudflare, Inc.*, which should protect the application, e.g., against *flooding*[21] resulting in Denial of Service (DoS). The `testing` and `staging` deployments of *SCA Tool Application*, which are hosted only within the FAU Virtual Private Network (VPN), do not have a Web Application Firewall (WAF) or similar placed in front of their web application proxy. *Ory Kratos* does not provide rate limiting against credential brute forcing.

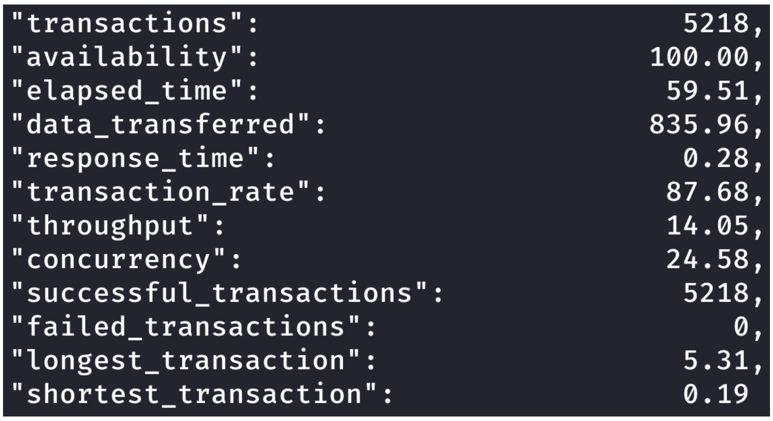
Adversaries with access to the FAU VPN can spawn *flooding*-attacks to exhaust the IT-infrastructure's resources or brute force credentials, and thereby hinder the developers during their work. Additionally, they can speed up the reconnaissance phase for future attacks against the `production` system, e.g., by performing path traversal or observing timing discrepancies[22].

[21] https://capec.mitre.org/data/definitions/125.html
[22] https://cwe.mitre.org/data/definitions/208.html

An HTTPS load test against the `production` deployment was performed from one single source IP address with *siege*[23] as shown in listing 3.17 to check if rate limiting in *Cloudflare* is configured. The test ran for 60 seconds and scored a transaction rate of 87.68 transactions per second, which leads to the conclusion, that no throttling is enabled. The overall results are shown in figure 3.9.

```
siege https://app.scatool.com -t 60s
```

Listing 3.17: *siege* Command



```
"transactions":                        5218,
"availability":                      100.00,
"elapsed_time":                       59.51,
"data_transferred":                  835.96,
"response_time":                       0.28,
"transaction_rate":                   87.68,
"throughput":                         14.05,
"concurrency":                        24.58,
"successful_transactions":             5218,
"failed_transactions":                    0,
"longest_transaction":                 5.31,
"shortest_transaction":                0.19
```

Figure 3.9: Results of HTTPS Load Test With *siege*

3.11 Missing IP Filtering

*Affected commit **2a40b0c015f328a6401943e3938729de1be0c626** of **2025-03-10**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.11**.*

As of December 1, 2023, the University of Erlangen-Nuremberg (FAU) has at least 46,213 members (Friedrlch-Alexander-University Erlangen-Nuremberg, 2023) plus external people with access to the FAU VPN. Consequently, the threat posed by attackers within the VPN cannot be overlooked. In addition to genuinely malicious attackers, this group may also include students from the Faculty of Engineering who are motivated by mere curiosity and a drive for scientific research to probe the VPN and test software tools against servers. These individuals can be categorized as script kiddies. Nevertheless, even a script kiddie can incapacitate deployments with DoS-attacks from a single client. There is a lack of filtering that allows only requests from authorized FAU members respectively *SCA Tool* members to reach the deployments.

[23] https://www.joedog.org/siege-home/

3.12 Insufficient Backup Strategy

*Finding recorded on **2024-12-24**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.12**.*

Currently, *SCA Tool* does not possess a formalized backup policy for application data or source code, although there are preliminary plans to establish one. At present, three distinct versions of the *SCA Tool Application's* source code exist: `production`, `staging`, and `testing`, whereby the two latter are stored in a *GitHub* repository. All versions are hosted at the Regionales Rechenzentrum Erlangen (RRZE). The *GitHub* repository is synchronized to the developers' end devices in non-monitored irregular intervals.

Application data is not synchronized between the three separate versions and are only backed up in the `production` environment to a different node on the same server.

In terms of security, each version of *SCA Tool Application* is secured by being stored on a VM that requires access via VPN or direct connection, thereby limiting exposure to unauthorized access. Access rights to these copies are predominantly granted to the core team, which comprises four members, with additional access provided to students involved in temporary projects.

One member of *SCA Tool* takes the responsibility for backup and recovery procedures, and application data backups are tested on an occasional basis.

Overall, it is evident that the existing backup framework lacks the necessary formality and comprehensiveness.

Cloned repositories on developers' devices are no real backups. The backups and the devices are not monitored by the IT administration and therefore the restoration and integrity cannot be tested orderly. They can be deleted anytime by the developers, they do not cover application data, customer data or the *GitHub* wiki. If they are stored on a movable end device, such as a notebook, they may be at the same geolocation as the other backups, e.g., during a meeting, and may get lost or stolen on public transport. The devices cannot be considered offline backups, since the devices are connected to the internet as soon as they are turned on. The copy may get destroyed by a computer virus before the notice about the need for the backup for restoration reaches the owner of the device.

The source code of *SCA Tool Application* as well as its documentation is stored at *GitHub*. *GitHub* is not a backup solution, but a platform for collaborative software development, which follows the *Shared-Responsibility-Model*[24]. GitHub, Inc. (2025) states in their Terms of Service (ToS), that while they are responsible for

[24] https://media.defense.gov/2024/Mar/07/2003407863/-1/-1/0/CSI-CloudTop10-Shared-Responsibility-Model.PDF

the operation of the platform and the provided services, the user is solely responsible for their content including its secure access and backup. Furthermore, *GitHub, Inc.* reserves the right to delete user content if it violates their policies. By using the services, the user agrees to them.

The other copies are all stored at the same datacenter, which makes them prone to elementary hazards, such as fire, earthquake, flood, etc. or unlucky circumstances, such as a plane crashing into the building or structural collapses. All copies are stored on Hard Disk Drives (HDDs), which may expose them to storage medium specific attack vectors.

The current approach raises concerns regarding the absence of a stringent policy, insufficient synchronization between backups, and limited testing of recovery procedures. Addressing these issues will be essential to bolster the organization's data integrity and resilience against potential data loss.

The geographical concentration has already been identified as a non-viable long-term solution. Therefore, *SCA Tool* considers the development of a cross-team strategy aimed at implementing dedicated cloud storage solutions in alternative geolocations to enhance redundancy and disaster recovery capabilities.

3.13 Missing Policy About Storage Encryption

Finding recorded on ***2025-01-16****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.13****.*

Some members of *SCA Tool*, e.g., students, use their private end devices to develop *SCA Tool Application*. They do not receive the directive to encrypt their disk. If their device gets stolen or lost, the source code of *SCA Tool Application* plus credentials in plain text, emails, etc. can be accessed by unauthorized people.

3.14 IT Service Providers Located Outside of EU

Finding recorded on ***2025-02-13****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.14****.*

Most parts of *SCA Tool Application* are hosted on servers provided by the RRZE, a central facility of the FAU. The following additional IT service providers used by *SCA Tool* are located in the United States of America (USA):

- Discord, Inc.[25]

[25] https://discord.com/

- Cloudflare, Inc.[26]
- GitHub, Inc.[27]
- Google[28]

Discord, Inc. provides the messaging platform *Discord*, which is used internally for communication between the team members of *SCA Tool*. *Cloudflare, Inc.* delivers solutions, such as domain registration, nameserver and WAF for *SCA Tool Application*. *GitHub, Inc.* offers the software code collaboration platform *GitHub*, where the source code of *SCA Tool Application* is stored. *Google* supplies *Gmail*, the email server, which is used for communication between *SCA Tool* and their customers as well as notifications by the *SCA Tool Application*.

Outsourcing IT services to the USA is a risk for data protection and reliability of service. At the latest since the second inauguration of *Donald J. Trump* as President of the USA on January 20, 2025, securities are no longer given.

Puglierin et al. (2025) suggest in their report for the *European Council on Foreign Relations (ECFR) e. V.*[29] that, in light of the latest political changes with *Donald J. Trump's* second term, the European Union (EU) might be considering a more autonomous approach to its international relationships, particularly with the United States (US). The changing sentiment among European countries, particularly in their views of the US as a necessary partner, rather than a clear ally, indicates a potential retraction from reliance on U.S. assets or influence. While it does not outright state that the EU is systematically pulling assets from the USA, it implies a shift towards more independent decision-making and partnership dynamics.

Netherlands Court of Audit (2025) indicates that the central government has begun utilizing cloud services without fully assessing the consequences, resulting in inadequate oversight of its cloud operations. They regard the central government's reliance on cloud solutions with concern. The risks to services offered to citizens and businesses, alongside the continuity of government operations, are excessive. The examined three public cloud contracts do not sufficiently uphold the principles of data protection, business continuity and digital sovereignty. Given that many cloud services are supplied by non-EU firms, this issue must be understood within the context of current geopolitical tensions.

Google's transparency report[30] clearly shows the requests made by US intelligence agencies on the basis of national laws.

[26] https://www.cloudflare.com/

[27] https://github.com/

[28] https://www.google.com/

[29] https://ecfr.eu/

[30] https://transparencyreport.google.com/user-data/us-national-security

The CLOUD Act[31], which stands for Clarifying Lawful Overseas Use of Data Act, is a US law that permits authorities to access all corporate and customer data from cloud and communication providers if the company is based in the US or falls under US jurisdiction.

Executive Order 12333[32] provides the National Security Agency (NSA) with the authority to collect, retain, analyze, and disseminate foreign signals intelligence information, primarily focusing on communications of foreign individuals outside the US. It also allows for the collection of communications involving US persons if they are communicating with foreign individuals. This collection is largely conducted globally and is not subject to regulation under the Foreign Intelligence Surveillance Act (FISA). Intelligence operations under EO 12333 must comply with specific minimization procedures approved by the Attorney General and established by the Secretary of Defense.

Section 702[33] of the FISA allows the Intelligence Community to collect targeted foreign intelligence information related to national security threats, focusing on non-US persons located outside the USA.

As Kuketz (2024) summarizes, the fundamental rights of European citizens can be potentially undermined by the extensive access rights of US authorities.

3.14.1 Discord

*Finding recorded on **2025-02-13**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.14.1**.*

The developer team behind *SCA Tool* uses the messenger *Discord*[34] for team communication. Discord, Inc. (2025) advertises that their messenger *Discord* is used by over 200 million monthly active users, being a popular messenger, especially among gamers and communities. The company behind *Discord*, *Discord, Inc.*, is a US corporation incorporated and registered under the laws of the State of Delaware, USA, and located in San Francisco, CA 94107, USA.

The context of data processing is defined as follows and thus the General Data Protection Regulation (GDPR) is interpreted accordingly as in the following sections. Members of the *SCA Tool* development team, i.e., students, doctoral candidates, etc., are data subjects. Their personal data is collected by the FAU, the data controller. For this purpose, the FAU provides the data subjects with a so-called *Discord* server. A *Discord* server is a virtual environment where users can join and engage with one another through text, voice, and video communica-

[31] https://www.justice.gov/criminal/media/999391/dl?inline

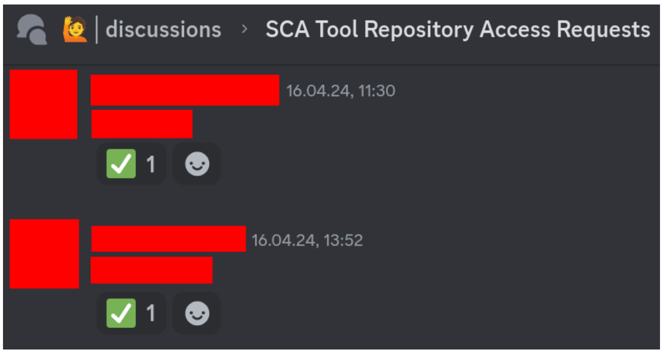
[32] https://www.archives.gov/federal-register/codification/executive-order/12333.html

[33] https://www.intel.gov/foreign-intelligence-surveillance-act/1237-fisa-section-702

[34] https://discord.com/

tion. This *Discord* server is not publicly accessible and is intended exclusively for *SCA Tool* development. The infrastructure of the *Discord* server is provided by *Discord, Inc.*, whereby *Discord, Inc.* acts as a data processor on behalf of the FAU. Even if the data subjects use the same *Discord* accounts on their own or other *Discord* servers, and therefore the data subjects' direct consent to data processing towards *Discord, Inc.* exists, the legal obligation of the FAU towards the data subjects under the GDPR must be maintained. In addition, while relaxing the requirements for data protection, Art. 89 GDPR at the same time guarantees data subjects data protection in the scientific environment.

Discord, Inc. (2024) state in their *Privacy Policy* the following. *Discord, Inc.* collects account information, content created by its users, payment information, information from users' actions, information from optional features, and other information, which the users provide directly to *Discord, Inc.* Additionally, *Discord, Inc.* accumulates information automatically about the users' devices, the use of the apps or the websites, and other information, e.g., users' actions when clicking on a referral link. Furthermore, *Discord, Inc.* gathers information from other sources, such as social media accounts, and connects these data to already collected ones. As shown in figure 3.10, data subjects were listing their *GitHub* usernames to get access to the *SCA Tool GitHub* repository. Through this initially seemingly harmless and justified query of *GitHub* usernames, users enable *Discord, Inc.* to create extensive profiles about them.



discussions › SCA Tool Repository Access Requests

[redacted] [redacted] 16.04.24, 11:30
[redacted]
✅ 1

[redacted] [redacted] 16.04.24, 13:52
[redacted]
✅ 1

Figure 3.10: *Discord*: GitHub Usernames

Discord, Inc. discloses users' information with the users' instructions, e.g., when they send messages to other users. They communicate users' data with their vendors, to comply with the law, in emergencies, e.g., to prevent substantial harm to an individual, to enforce their policies and rights, such as their ToS or their community guidelines. *Discord, Inc.* shares users' information with their related companies. Furthermore, users' data may be disclosed during selling, acquiring or transferring assets. Moreover, aggregated or de-identified users' information may be disclosed to partners or the public.

In their *Data Retention Policy* of *Discord*, Discord, Inc. (2023) state, that data, which were shared with other users, such as messages, images, etc., will be retained when a user's account gets deleted.

[redacted] [redacted] 16.04.24, 11:25

[redacted]

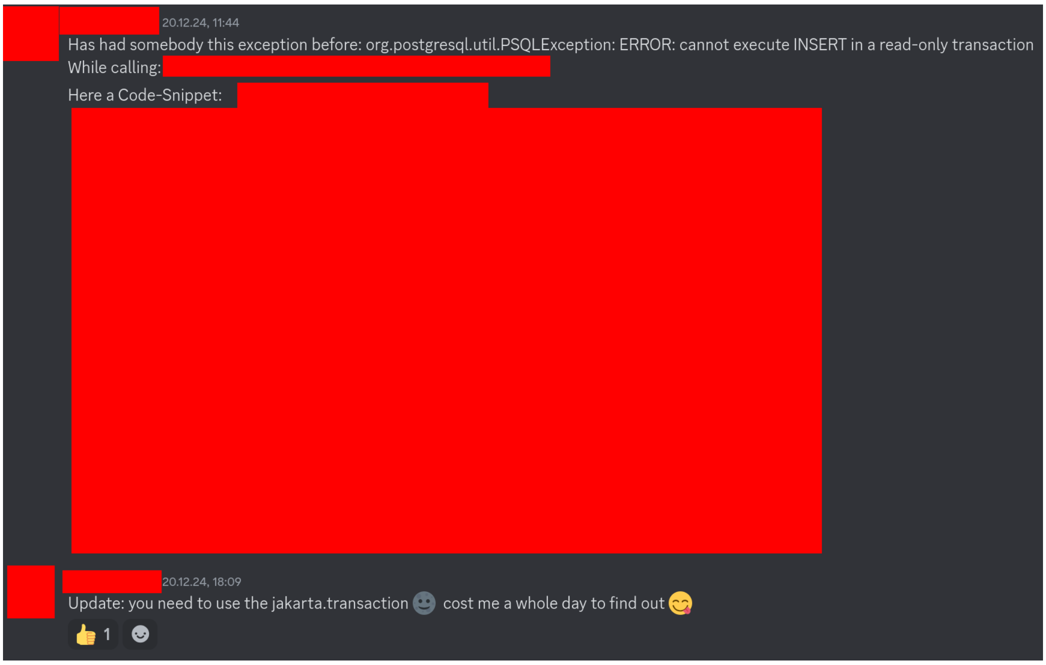
[redacted] [redacted] 16.04.24, 13:43

[redacted]

Figure 3.11: *Discord*: PII

Discord is not OSS. While this fact does not necessarily mean that it is more insecure than if it was OSS (Schryen & Kadura, 2009), its source code cannot be reviewed by independent security researchers. *Discord, Inc.* is also not certified by *ISO/IEC 27001* or others, such as System and Organization Controls 2 (SOC 2), a framework to assess data security and privacy controls in service organizations. This means that there is no reliable source that guarantees the security of *Discord*. While providing End-to-End (E2E)-encryption for audio and video calls. Messages, images, etc. are not E2E-encrypted, but only encrypted in transit via TLS. Post-quantum encryption is not implemented as well. Contacts cannot be verified and users do not get notified if the contact's fingerprint changes. The data residency can also not be chosen. Furthermore, it is not possible to conclude a Data Processing Agreement (DPA) with *Discord, Inc.* These points are against Art. 28 GDPR, which mandates the data controller to use data processors offering adequate assurances to implement suitable technical and organizational measures. This ensures that the processing complies with the standards set out in the GDPR and protects the rights of the data subject. Furthermore, a DPA between data controller and data processor is obligatory. (European Parliament and Council of the European Union, 2016)

As shown in figures 3.12 and 3.13, source code and the system model view of *SCA Tool* are published on *Discord*. Organization secrets should never be uploaded to an untrusted platform, that evaluates these information and shares them with other companies.



[redacted] [redacted] 20.12.24, 11:44

Has had somebody this exception before: org.postgresql.util.PSQLException: ERROR: cannot execute INSERT in a read-only transaction

While calling: [redacted]

Here a Code-Snippet: [redacted]

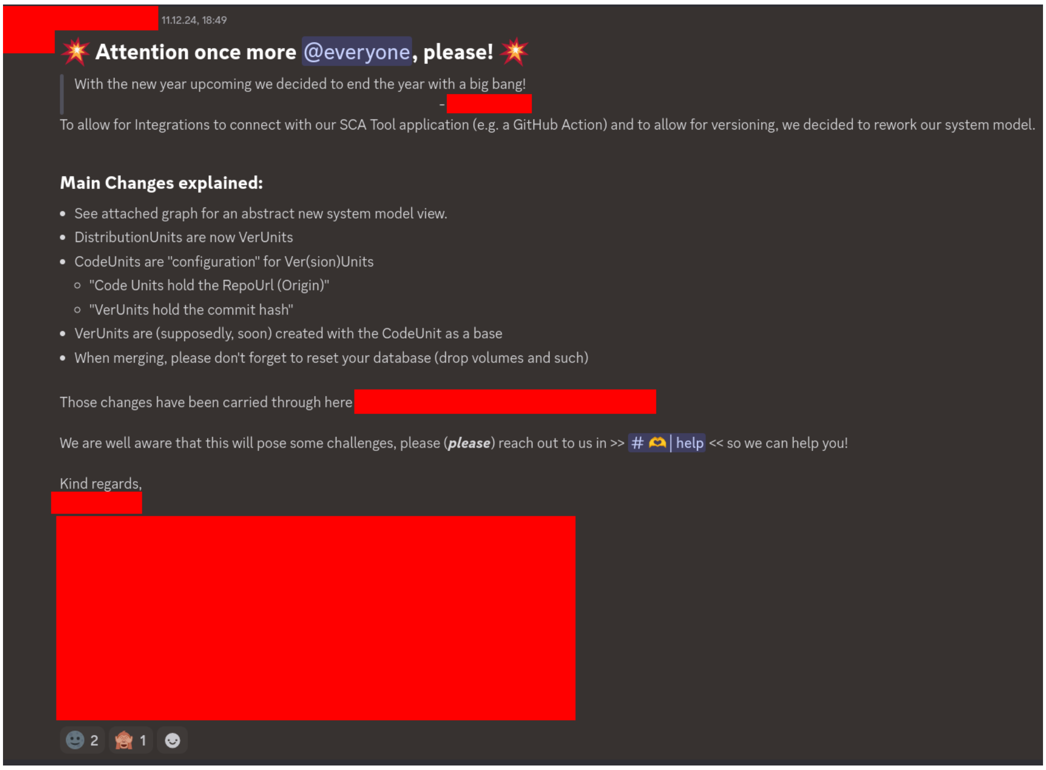
[redacted]

[redacted] [redacted] 20.12.24, 18:09

Update: you need to use the jakarta.transaction 😕 cost me a whole day to find out 😋

👍 1

Figure 3.12: *Discord*: *SCA Tool Application* Source Code Snippet



[redacted] 11.12.24, 18:49

💥 Attention once more @everyone, please! 💥

> With the new year upcoming we decided to end the year with a big bang!

- [redacted]

To allow for Integrations to connect with our SCA Tool application (e.g. a GitHub Action) and to allow for versioning, we decided to rework our system model.

Main Changes explained:

- See attached graph for an abstract new system model view.
- DistributionUnits are now VerUnits
- CodeUnits are "configuration" for Ver(sion)Units
  - "Code Units hold the RepoUrl (Origin)"
  - "VerUnits hold the commit hash"
- VerUnits are (supposedly, soon) created with the CodeUnit as a base
- When merging, please don't forget to reset your database (drop volumes and such)

Those changes have been carried through here [redacted]

We are well aware that this will pose some challenges, please (***please***) reach out to us in >> # 🫶 | help << so we can help you!

Kind regards,
[redacted]

[redacted]

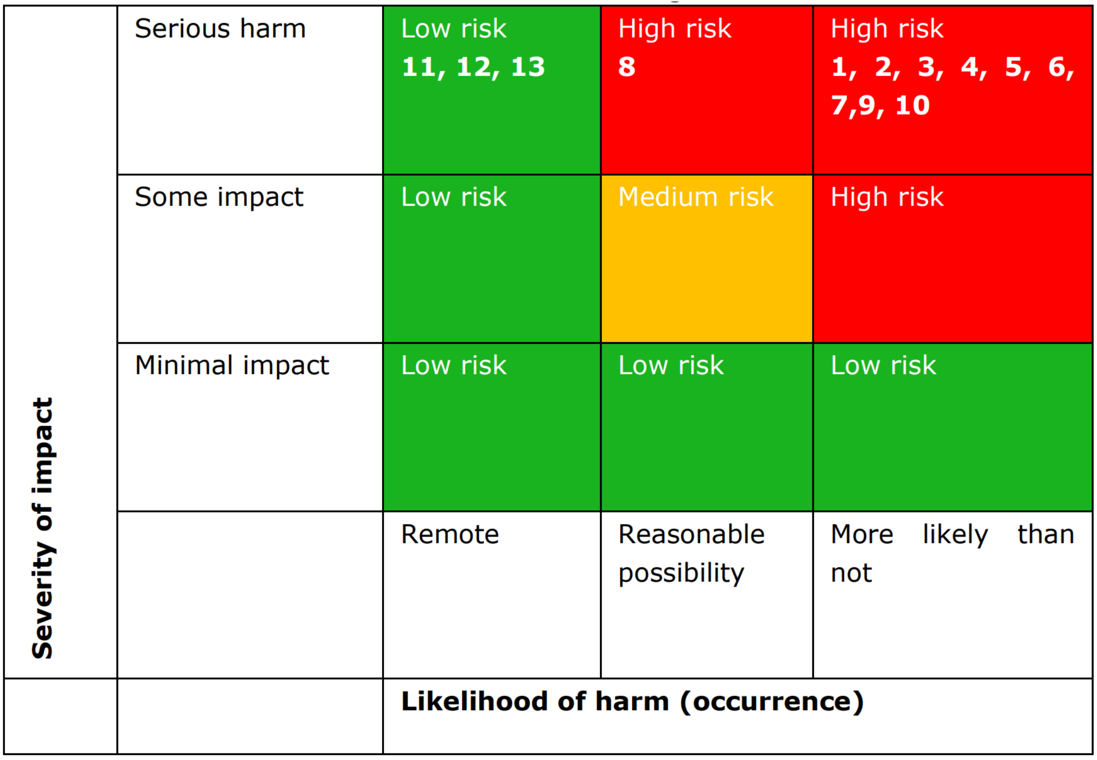
😕 2 🙈 1

Figure 3.13: *Discord*: *SCA Tool Application* System Model View

3.14.2 Google

Finding recorded on ***2025-02-13****.*

In the name of the Dutch *Ministry of Justice and Security*, Nas and Terra (2021) assessed the impact for data protection by *Google Workspace*, a cloud service combining multiple *Google* software services including but not limited to *Drive*, *Calendar*, *Gmail*, and *Contacts*. The senior advisors of *Privacy Company*[35] identified 13 risks, which can result in one or more of loss of confidentiality, loss of control, unlawful processing, unlawful further processing, risk of reidentification, and chilling effects to exercise (related) rights. The single risks are categorized by their severity of impact times their likelihood of occurrence and graphically presented in figure 3.14.



Severity of impact	Serious harm	Low risk **11, 12, 13**	High risk **8**	High risk **1, 2, 3, 4, 5, 6, 7,9, 10**
	Some impact	Low risk	Medium risk	High risk
	Minimal impact	Low risk	Low risk	Low risk
		Remote	Reasonable possibility	More likely than not
		Likelihood of harm (occurrence)		

Figure 3.14: Risk Matrix for *Google Workspace* (Nas & Terra, 2021)

The ten high risks were addressed to *Google*, who mitigated two of them to the fullest and four significant risks partly. Consequently, eight high risks remained even after the mitigation efforts were implemented.

This fact, combined with the theoretical possibility of abuse of the law by the US government, leads to the conclusion that the risk for *SCA Tool* is not acceptable

[35] http://www.privacycompany.eu/

and that one or more European alternatives to *Google Workspace* should be considered. It is not only the protection of the data subject that is important here, but also the protection of *SCA Tool* with regard to possible legal violations with the associated efforts and penalties.

3.14.3 Cloudflare and GitHub

Finding recorded on ***2025-02-13****.*

Cloudflare, Inc. and *GitHub, Inc.* may not be suitable IT service providers for *SCA Tool.*

Dr. Johann Schlamp et al. (2022) describe several outage scenarios that could affect the transatlantic cables between Europe and America. The causes are DoS, cable damage, human error, peering dispute, software failure, state actions, and technical defects. Without a failover[36], such incidents could disrupt the accessibility of the *SCA Tool Application* for an indefinite period of time.

Furthermore, both providers allow setting the data residency in EU, but based on the mentioned US laws and regulations, this cannot be equated with providers having their registered office in the EU. Currently, EU data residency is not configured for both providers. The risk for *SCA Tool* is low, especially since *Cloudflare* and *GitHub* are not the main PII data storing and processing entities in the *SCA Tool Application* environment.

The use of existing European alternatives, same functionalities presumed, would mitigate the risk of data protection lawsuits and could allow *SCA Tool* to continue their servicing for EU-members in case of mentioned outage scenarios. The highlighted risks do apply generally to the use of service providers located in the USA.

A comprehensive analysis of *Cloudflare, Inc.* and *GitHub, Inc.* regarding their secure configurations and compliance should be done in detail as stated in chapter 8.

3.15 Email Notifications via Domain *gmail.com*

Finding recorded on ***2025-02-15****.*
The mitigation strategy to this finding is detailed in (sub)section ***5.15****.*

Some emails sent to the users, such as for password recovery, are sent from `info.scatool@gmail.com` and not from the domain `scatool.com`, despite some emails, e.g., the `Welcome to SCA Tool!`-email, are sent from `scatool.com`.

[36] https://csrc.nist.gov/glossary/term/failover

While this does not only seem less professional and weakens the brand identity, it will also train the users to trust business emails from `gmail.com`. Consequently, phishing[37] attacks can be carried out much easier and more successfully by adversaries. In addition, email addresses provided by *Google's Gmail*[38] are free of charge, which allows attackers to spawn new email addresses if the spam reputation of used ones has gone bad. Listing 3.18 shows, with how little effort of new trustworthy-seeming email addresses can be thought.

```
noreply.scatool@gmail.com
support.scatool@gmail.com
sales.scatool@gmail.com
feedback.scatool@gmail.com
newsletter.scatool@gmail.com
app.scatool@gmail.com
scatool.info@gmail.com
scatoolapp@gmail.com
info.scatoolapp@gmail.com
noreply.scatoolapp@gmail.com
```

Listing 3.18: (Fictive) *Gmail*-Addresses not Possessed by *SCA Tool*

Furthermore, security-aware users, who do not trust professional emails sent from `gmail.com`, will less likely react to actually important emails, e.g., ones that are informing about unusual account behavior or asking for verification.

If the email provider shall be changed in the future, the use of *Google's* own domain hinders the transfer of the email address to the new provider. Additionally, changes in the source code of *SCA Tool Application* would be required. Moreover, users would have to adapt filters and automations in their IT-infrastructure to a new email address, which could disrupt existing processes and lead to frustration on the user's side.

3.16 Account Takeover

*Finding recorded on **2025-02-20**.*
*The mitigation strategies to this finding are detailed in (sub)sections **5.16** and **5.17**.*

If the user, referred to as the victim, changes their email address to one that does not belong to them, an adversary in possession of the new email address can gain access to the victim's account without the victim's notice.

For the description of an attack, the following prerequisites apply:

- Official domain of *SCA Tool* is *scatool.com*.

[37] https://attack.mitre.org/techniques/T1566/
[38] https://mail.google.com/

- One of the official email addresses is *info.scatool@gmail.com.*
- Victim ***V*** of organization *OrgV* with email address *v@orgv.com.*
- Adversary ***A*** with email address *info.scatoolapp@gmail.com* and in possession of the domain *scatoolapp.com.*

The attack could be carried out as described in the following steps:

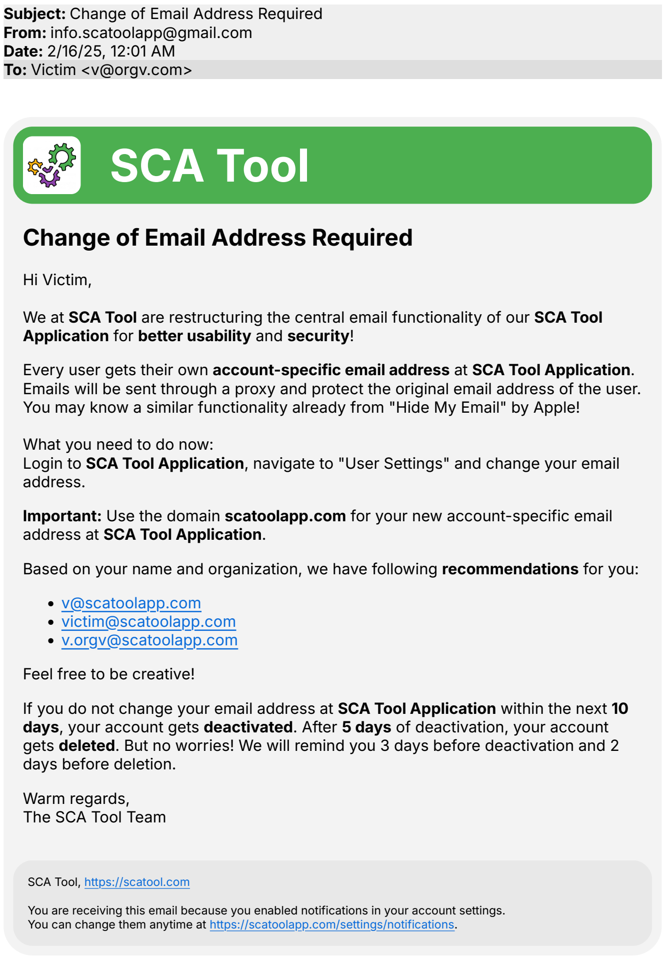
1. ***A*** sends a phishing[39] email from *info.scatoolapp@gmail.com* to ***V***, requesting them to change their email address from *v@orgv.com* to one under the domain *scatoolapp.com.* An example of a phishing email can be seen in figure 3.15.
2. ***V*** considers the phishing email to be authentic and follows the given instructions by logging into their account of *SCA Tool Application.*
3. ***V*** opens the `User Settings` and changes their email address to the freely chosen email address *v@scatoolapp.com.*
4. ***A*** receives an email from the official email address *info.scatool@gmail.com* with the request to verify the ownership of *v@scatoolapp.com.* ***A*** does not verify.
5. ***A*** opens the login page of *SCA Tool Application* and clicks on the link `Forgot your password?`.
6. ***A*** requests a recovery code for *v@scatoolapp.com*, receives the recovery code and recovers the account. (***A*** may need to add the URL-parameter `refresh=true` to continue to the `workspace`.)
7. ***A*** changes the password as demanded by *SCA Tool Application*, which gives them full control over ***V****'s* account. ***V****'s* session remains active.
8. ***A*** eavesdrops on ***V***. ***V*** is not aware of this. None of the sessions of ***A*** and ***V*** are being terminated.
9. ***V*** logs out and cannot log in anymore due to the password change by ***A***. ***A*** is in sole control over ***V****'s* account.

During this attack, no email is sent to ***V*** informing them about the change of their email address. Only if ***V*** logs out and tries to log in again, they will recognize, that they cannot access their account. If ***A*** logs out, ***V****'s* session will remain active. ***A*** can log out and log in as often as they want without affecting ***V****'s* session.

While the described phishing scenario is actively targeting the user, the user themselves can be the cause of this issue as well. If the user wants to change

[39] https://attack.mitre.org/techniques/T1566/

their email address, they may accidentally misspell it. If they do not recognize their mistake and click the `Save`-button, the owner of the misspelled email address receives the verification request. The owner could report it to *SCA Tool* or simply ignore the email. By this, ***V*** would only lose access to their account as soon as they log out, but they would not receive damage. Their access to the account could be restored by *SCA Tool* after authentication of ***V*** via other means. In contrast to that, the owner could proceed with malicious (or just curious intentions) as described from step 5 on. This also proves, that no particularly high technical skills are required to carry out such an attack.



Subject: Change of Email Address Required
From: info.scatoolapp@gmail.com
Date: 2/16/25, 12:01 AM
To: Victim <v@orgv.com>

SCA Tool

Change of Email Address Required

Hi Victim,

We at **SCA Tool** are restructuring the central email functionality of our **SCA Tool Application** for **better usability** and **security**!

Every user gets their own **account-specific email address** at **SCA Tool Application**. Emails will be sent through a proxy and protect the original email address of the user. You may know a similar functionality already from "Hide My Email" by Apple!

What you need to do now:
Login to **SCA Tool Application**, navigate to "User Settings" and change your email address.

Important: Use the domain **scatoolapp.com** for your new account-specific email address at **SCA Tool Application**.

Based on your name and organization, we have following **recommendations** for you:

- v@scatoolapp.com
- victim@scatoolapp.com
- v.orgv@scatoolapp.com

Feel free to be creative!

If you do not change your email address at **SCA Tool Application** within the next **10 days**, your account gets **deactivated**. After **5 days** of deactivation, your account gets **deleted**. But no worries! We will remind you 3 days before deactivation and 2 days before deletion.

Warm regards,
The SCA Tool Team

SCA Tool, https://scatool.com

You are receiving this email because you enabled notifications in your account settings.
You can change them anytime at https://scatoolapp.com/settings/notifications.

Figure 3.15: Phishing Email

In addition to alterations to user accounts, sensitive information can be extracted. This encompasses not only ***V****'s* email address and organizational affiliation but also the details of other users who belong to the organization, which include their names and email addresses. Additionally, sensitive data such as commit messages, CVEs, and the libraries used in the scanned software projects can be accessed. Furthermore, the account may be exploited for spam distribution by incorporating users external to the organization through the `User Invite`-function. These external users would also gain access to the sensitive information.

Moreover, ***V's*** organization could face direct and indirect financial repercussions. ***V's*** organization could be subject to extortion using the acquired trade secrets, and the disclosure of such information could potentially destroy their business. It could severely damage the reputation among customers if sensitive data, such as the CVEs, are made public, and could lead to successful cyberattacks on their customer's side. Additionally, addressing all these issues would entail administrative expenses.

3.17 Insufficient GitHub Access Control

*Finding recorded on **2025-02-20**.*
*The mitigation strategy to this finding is detailed in (sub)section **5.18**.*

The source code repository of *SCA Tool Application* is stored at *GitHub*. Members of *SCA Tool* are associates of the FAU, which implies that they are registered in the Identity Management System (IdMS) of the FAU. This includes access to the FAU VPN and their own FAU email address.

Currently, no MFA is enforced for members of the organization. If credentials of the developers get leaked, no second factor prevents unauthorized access to the source code repository. Additionally, no filtering by originating IP address happens. This means, that if adversaries are in control of the credentials, they can access the source code from everywhere in the world, which can also lead to an increased number of unauthorized accesses if the credentials are published on the internet. Additionally, if the members of *SCA Tool* are connected to an insecure network, such as spanned up by a public Wireless Fidelity (Wi-Fi) hotspot, without securing their connection with an encrypted VPN-connection, they are more prone to AiTM-attacks.

Furthermore, email notifications are not bound to specific domains. Emails about the organization or the source code will be sent to any domain, including private email accounts. Their security cannot be guaranteed by *SCA Tool*, which could lead to a potential leak of organizational information if unauthorized access to a member's email account happens or the connection between *GitHub's* email server and a member's email server or between a member's email server and their email client are not encrypted.

GitHub allows the restriction of access by IP address whitelisting. By this, access to the repository is only granted to users, whose requests originate from the predefined IP address ranges. As mentioned earlier, this setting is not applied.

4 Design of an ISMS

This chapter first outlines the considerations that were made when establishing an ISMS. Subsequently, the developed policies will be presented.

4.1 Strategies for Designing an Effective ISMS

Figure 4.1 shows the three variants of the *IT-Grundschutz* methodology in a graphical illustration. The basic protection approach is designed for organizations seeking to initiate their engagement with *IT-Grundschutz*, facilitating the prompt securing of all pertinent business processes at a fundamental level. The core protection variant is concentrated on safeguarding an organization's essential assets and critical business operations, thereby aiming to provide comprehensive protection for the most vital components. In contrast, the standard protection aligns with the established *IT-Grundschutz* methodology, thereby representing a recommended approach for organizations aiming to enhance their security posture.

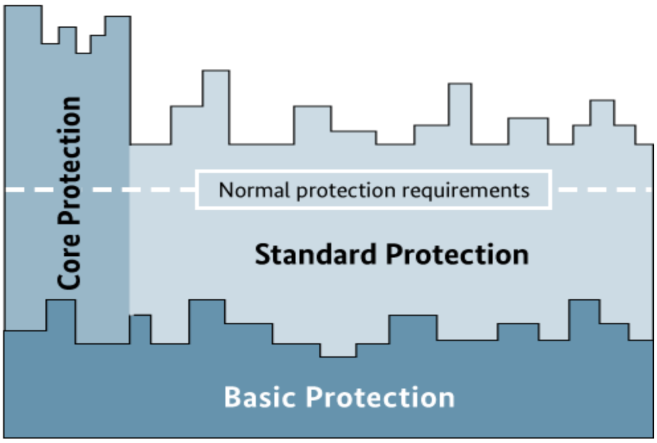


Figure 4.1: *IT-Grundschutz* methodology (Fraunhofer SIT, 2018, Figure 5)

When developing policies, establishing rules, or implementing security mechanisms, it is essential to consider the current circumstances of the organization and its surrounding environment. For instance, the political landscape significantly influences these decisions. Recent changes in the political climate have rendered service providers based in the USA potentially unreliable partners for the future. As a result, it is advisable for future service providers to be located within the EU to ensure compliance with stringent data protection regulations and to enhance future reliability. Moreover, there is an increasing necessity to implement robust security mechanisms against various attack techniques, including *shoulder surfing*. A simple, lengthy password displayed in plaintext format is insufficient to protect sensitive information. Traditionally, shoulder surfing referred primarily to the act of an attacker physically peeking over the user's shoulder. However, in contemporary contexts, there are numerous recording devices, such as smartphones, smart glasses, webcams, and surveillance cameras, which surround the user. These devices not only capture any credentials that are displayed but can also retain this information for extended periods. Furthermore, such recordings have the potential to be disseminated across social media platforms, reaching millions of viewers. In summary, it must be stated that dangers previously perceived as less problematic need to be reassessed continuously due to technological advancement.

Organizations as well as their environment can evolve over time. As they pursue new areas of responsibility, change their employee structure, or modify their IT landscape, it may become necessary to reassess and potentially expand policies, processes, and procedures. Therefore, regular review and possible adjustments are essential, as previously described in section 2.1. It is crucial that changes are communicated to all members of *SCA Tool* thoroughly and clearly. This can be achieved, for example, through a monthly organizational newsletter.

The *Professorship for Open-Source Software* manages several projects next to *SCA Tool*. While these projects operate independently, there are overlaps in the utilization of the same IT, human, and other resources. Therefore, it is advisable to implement *Core Protection* according to the *IT-Grundschutz* within each individual project, while pursuing *Basic Protection* across the professorship. Over time, the protection level within the professorship can be elevated to *Standard Protection*.

The following policies are defined based on the findings identified in chapter 3 and take requirements of the *IT-Grundschutz Compendium* into account. For better readability and further use, the requirements are not marked as citations but referenced at the end of each policy. The policies establish a foundation for an ISMS without overwhelming *SCA Tool* with excessive requirements. This approach aims to preserve the agility of the young organization. The members of *SCA Tool* are advised to read the references in order to adjust the policies based on the organization's needs and to raise their security awareness.

4.1.1 Context of the Organization

In order to define and implement the objectives of the ISMS, it is necessary to understand the context of the organization. One must distinguish between internal and external factors. External factors include, for example, global political, legal or cultural aspects, while internal factors include, for example, the general corporate culture or the strategic orientation of the company.

The organization *SCA Tool* is a project by the *Professorship for Open-Source Software* at FAU. They are considered a small enterprise and are providing the *SCA Tool Application* to their customers, which allows them to scan their software projects for license and vulnerability issues. The customers to target as well as their size are defined, but won't be published in this thesis. There is no limitation in the number of targeted customers. The customers are operating in all sectors worldwide, in which open source software is used. The general corporate culture of *SCA Tool* (e.g., clan, adhocracy, market, hierarchy) as an internal factor is not yet defined. External factors, such as legal aspects, political aspects, and cultural aspects were not considered yet.

With the information above, policies are created tailored to *SCA Tool* and their needs in order to implement IS efficiently.

4.1.2 Roles and Responsibilities

To successfully implement policies within an organization, roles and their responsibilities should be clearly defined and documented. These roles should be independent of individual members of *SCA Tool* and should be defined solely by their function within *SCA Tool*. Processes then assign tasks and assets to these roles for which they are responsible based on superordinate policies.

An organizational chart can assist in illustrating dependencies among stakeholders, optimizing communication pathways, and promoting better collaboration. Particularly for new members of an organization, an organizational chart can facilitate onboarding and provide a clearer understanding of the organizational context.

The creation of an organizational chart as a subtask of this Master's thesis is not advisable. Instead, it should be developed by an individual who has the appropriate overview. It is advisable to establish this organizational chart for the entire professorship, enabling cross-project communication and fostering effective collaboration among projects.

4.2 Information Security Management Policy

Purpose

This policy represents the highest directive in *SCA Tool* for Information Security (IS), setting a basic level of security. It describes the necessary measures to ensure the confidentiality, integrity and availability of company information across all business processes and projects. The security of the web application *SCA Tool Application*, which is developed by *SCA Tool*, and the associated business processes and their continuity are to be guaranteed with this policy.

Scope

This policy affects all members of *SCA Tool* while paying special attention to the upper management. It is to be applied to all business processes and projects of *SCA Tool*.

Definitions

None.

Policy Statement

This policy sets the foundation to Information Security Management (ISM) by introducing the following rules:

Commitment by the Head of *SCA Tool*
The Head of *SCA Tool* must commit to IS. They must assign roles and responsibilities to ensure the process of IS and provide the necessary resources. They must be informed regularly about the current state of IS, risks and consequences of missing security measures.

Information Security Management System
There must be an Information Security Management System (ISMS), which defines the systematic approach to manage the IS of *SCA Tool* and to ensure the confidentiality, integrity, and availability of information.

Audits and Revisions
The ISMS must be audited annually and revised in a timely manner to changing business scopes and processes.

Communication
The IS objectives and strategies, which are defined in the ISMS, must be clearly communicated to all employees. The importance of compliance must be emphasized. Employees must be included in the process of IS.

Roles and Responsibilities
The Head of *SCA Tool* must appoint an Information Security Officer, who reports to the Head of *SCA Tool* at regular intervals and provides recommendations for implementation of security measures. They must be provided with necessary resources and be included in all IS-relevant processes and projects. In addition, a Data Protection Officer must be nominated, to ensure the compliance with data protection laws.

Compliance With the Law and Regulations
SCA Tool must act in accordance with the law and follow current best practices of IS.

Adequate Information Security Management
Investments in IS must be made homogeneously and coordinated across entire *SCA Tool*. This includes financial investments as well as organizational and personnel ones.

Documentation
IS-requirements, the process, and applied measures must be documented.

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: ISMS.1 Sicherheitsmanagement

4.3 Secure Software Development and Operation Policy

Purpose

The purpose of this policy is to establish guidelines for secure software development and operation of *SCA Tool Application*. The main goals are to provide the secure usage of *SCA Tool Application* and to protect the data it is processing.

Scope

This guideline applies to all developers, IT administrators and other persons who have a responsibility in the software development of *SCA Tool Application*.

Definitions

None.

Policy Statement

The **software development** of *SCA Tool Application* must fulfill the following requirements:

Documentation of Source Code Changes
Any change to the source code of *SCA Tool Application* must be documented including date, time, author, and change.

Authentication
If parts of *SCA Tool Application* shall only be available to a specific group of users, the users must authenticate themselves. The authentication process must follow current best practices.

Storing Source Code
Source code of *SCA Tool Application* must only be stored and processed on encrypted storage media or media, which are secured against unauthorized access by other means.

Documentation of Source Code
Source code, which is not self-explanatory, must be documented. The documentation language must be consistent across *SCA Tool Application*.

Management of Software Development
Software development must be managed in such a way, that the prioritization of security-relevant bugs or functions does not open up risks for the confidentiality, integrity and availability of *SCA Tool Application*.

Mitigation of Security Risks

Software developers must mitigate the most critical security risks for *SCA Tool Application*. To achieve this, they must inform themselves about potential risks and requirements through relevant literature and apply measures if applicable. The following documents should be considered in this context:

- OWASP Top 10
- *IT-Grundschutz Compendium*: CON.10 Entwicklung von Webanwendungen

Recruitment of Qualified Software Developers

Software developers to be hired must have demonstrable software development skills, including knowledge about secure software development. If this knowledge is not existent, the software developer must undergo a training about secure software development beforehand.

Regular Testing

SCA Tool Application must be tested at regular intervals manually or with automated tools to detect issues affecting its security or main functionality.

External Software or Services

If services are included in *SCA Tool Application*, the contracts with external service providers must be sufficiently designed. External software must only be included in *SCA Tool Application* if the licensing model is not restricting the intended direction of license of *SCA Tool Application*. Contracts and licenses must be respected at all times.

Regarding the **operation** of *SCA Tool Application*, following criteria must be met:

Updates

SCA Tool Application as well as its underlying hardware must have the latest security patches applied. If a latest security patch is breaking the functionality of *SCA Tool Application*, a mitigation must be defined. If no mitigation is possible, the underlying issue must be documented and closely monitored until a mitigation is possible.

Logging of Security-Relevant Events

Security-relevant events must be logged. The logged events must be monitored and evaluated to apply countermeasures in case of undesired behavior.

Hiding of Security-Relevant Information

Security-relevant information about *SCA Tool Application*, e.g., versions of used software, must not be published.

Protection Against Automated Misuse
Measures must be applied to *SCA Tool Application* in order to protect against automated misuse, such as Distributed Denial of Service (DDoS)-attacks or brute-force-attacks.

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- OWASP Top 10
- OWASP Cheat Sheet Series
- *IT-Grundschutz Compendium*: CON.8 Software-Entwicklung
- *IT-Grundschutz Compendium*: CON.10 Entwicklung von Webanwendungen
- *IT-Grundschutz Compendium*: APP.3.1 Webanwendungen und Webservices
- *IT-Grundschutz Compendium*: APP.7 Entwicklung von Individualsoftware

4.4 Identity and Access Management Policy

Purpose

In order to protect sensitive information, minimize security risks, and ensure compliance with regulatory requirements, access to valuable resources in an institution should be restricted to authorized users and IT components. This policy provides guidelines to ensure secure identity and access management.

Scope

This policy applies to all members and resources of *SCA Tool*.

Definitions

None.

Policy Statement

Following criteria are applicable to **user accounts and IT services as subscriber** to IT services:

Least-Privilege Principle
Identities and access rights must only be as privileged as required for the user to fulfill their work.

Need-to-Know Principle
Identities and access rights must only be given to the user based on their actual needs for fulfilling their work.

Monitoring of Access
Access to sensitive resources must be monitored and misuse documented.

Review of Access
Every six months, resources should be reviewed for granted permissions, that are no longer needed.

Revocation of Access
Access to resources must be revoked as soon as the user or IT service is no longer related to *SCA Tool*.

Reporting Suspected Compromise
As soon as there is suspicion that an account or an IT service within the *SCA Tool's* jurisdiction has been compromised, the affected user must immediately and directly report this suspicion to the responsible IT administrator of *SCA Tool*.

Access via VPN
The user must connect to the FAU VPN before accessing IT resources of *SCA Tool*.

Following criteria must be met by **IT services** of *SCA Tool* **as provider**.

Virtual Private Network
If the IT service does not need to be publicly accessible, and its nature and hosting environment allow it, then this IT service may only be accessible within a VPN. Access to the VPN must be secured with Multi-Factor Authentication. It is to ensure that all authorized users can connect to the VPN using different devices without significant additional effort or technical challenges.

Multi-Factor Authentication
If the IT service provides the IT administrator with the option to enforce MFA for the user, then it should be implemented after proper instruction of the user.

Rate Limiting
If the IT service allows it, rate limiting must be enabled for access to publicly available endpoints or login interfaces.

IP Address Filtering
If the IT service allows it, access should only be granted from defined IP address ranges. This does not apply if it renders the IT service unusable, significantly impairs its functionality, or denies access to legitimate users.

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: ORP.4 Identitäts- und Berechtigungsmanagement

4.5 Password Policy

Purpose

This policy describes rules and guidelines that govern the creation, complexity, handling, and management of credentials.

Scope

This policy applies to all members of *SCA Tool* and (IT) service accounts delivered by *SCA Tool* or by a third-party to the members of *SCA Tool*.

Definitions

Password

A password is a secret chosen by the user to be used as an authentication factor. A password can consist of letters, numbers, or special characters, e.g., punctuation or spaces. In this policy, password is a hypernym including password, passphrase, Personal Identification Number, secret.

Policy Statement

Following criteria must be met by users using corporate accounts:

Password Requirements for User Accounts

Users must create their passwords based on the following requirements:

- Minimum length of twelve characters
- Three of the following four properties:
  - At least one special character (e.g., !?<>[],;.:-_)
  - At least one uppercase letter (A-Z)
  - At least one lowercase letter (a-z)
  - At least one digit (0-9)
- Not based on personal information (e.g. date of birth, place of residence, first and last name, name of pet, ...) or information of *SCA Tool* (e.g., department name, founding year, function in *SCA Tool*, etc.).
- Does not contain a word in any language, dialect, jargon, slang, etc. (e.g., rainbow, LOVELIFE, sk8erboi).
- Is not included in a breached passwords list.

- Is unique and will not be used for any other account.

Password Requirements for Administrators

Passwords for IT administrator accounts must, in addition to the requirements for user accounts, meet the following criteria:

- Minimum length of 20 characters

Multi-Factor Authentication

Wherever possible, MFA must be set up and used by the user. During the creation of the second factor, it is possible that *backup codes* are created. These *backup codes* must be stored securely by the user and protected from unauthorized third parties, as they can be used in the event of the second factor is lost. They can be used once to gain access to the protected account.

Sharing Credentials

User-related credentials must not be shared.

Storing Credentials

Credentials must not be stored in online storage solutions in plain text. A password manager must be used to store credentials and be secured by two means, such as a password in combination with a key file. The master password must fulfill the password requirements stated above.

Transmission of Credentials

The digital transmission of credentials must occur with End-to-End encryption. This can be achieved by encrypting emails using Pretty Good Privacy (PGP) or Secure/Multipurpose Internet Mail Extensions (S/MIME), or by embedding the access data in a password-protected ZIP file. The password for the ZIP file must be transmitted through a second channel (e.g., by phone call or SMS).

Reporting of Lost or Exposed Credentials

As soon as there is suspicion that credentials were lost or exposed to non-authorized entities, the affected user must immediately and directly report this suspicion to the responsible IT administrator of *SCA Tool*. The password for the affected account or IT service must be promptly changed by the user in accordance with this policy. This also applies if personal devices (smartphones, computers, etc.) are hacked or manipulated, which contain access credentials to IT services of the *SCA Tool*.

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: ORP.4 Identitäts- und Berechtigungsmanagement
- NIST SP 800-63B-4 2pd

4.6 Backup and Recovery Policy

Purpose

This policy outlines the requirements for the regular backup of data and its recovery to protect against data loss, corruption, and disaster. The aim is to ensure business continuity and compliance with regulatory standards.

Scope

This policy applies to all information in digital and physical form, i.e., digital assets, databases, configurations, and customer data, that are critical to the operation of *SCA Tool*.

Definitions

None.

Policy Statement

Following criteria related to **backups** must be met, including the *3-2-1-1-0*-rule:

Number of Copies *(3)*
Three copies of data must exist.

Backup Media *(2)*
The copies must exist on at least two different media types (e.g., Solid-State-Drive (SSD) and tape storage) to protect against medium-specific threats. The backup media must be protected against unauthorized access.

Backup Location *(1)*
One copy must be stored off-site to protect against environmental disasters, e.g., fire, flood.

State of Copies *(1)*
At least one copy must be offline to protect against threats spreading through the network, e.g., Ransomware.

Error-Free Backups *(0)*
Integrity and successful restoration must be tested regularly to ensure no errors in backups.

Backup Duration
The backup process of data must be finished before the next backup process of the same data starts.

Documentation of Backups
The creation of backups must be documented, including date, time, size, status (success, error, ...).

Backup Frequency
Backups must be created frequently, depending on the data to be backed up:

- Daily Backups: For critical data (e.g., databases, essential applications).
- Weekly Backups: For important data (e.g., project files, business documents).
- Monthly Backups: For non-critical data (e.g., archives, old projects).

Backup Retention
If no legal regulations require differing backup retention periods, the following applies:

- Daily Backups: Retained for a minimum of 14 days.
- Weekly Backups: Retained for a minimum of 3 months.
- Monthly Backups: Retained for a minimum of 1 year.

Backup by User
The user is responsible for backing up their data if the underlying system is not backed up automatically, e.g., by transferring the data to a system, which is backed up automatically.

Data Protection
Data protection regulations, such as the General Data Protection Regulation (GDPR), must be complied with.

Secure Storing Backups
Backups must be stored securely and only be accessible to authorized members of *SCA Tool*.

Monitoring of Backup Process
The backup process must be monitored to detect possible issues at an early stage.

Following criteria related to **recovery** from backups must be met:

Roles and Responsibilities
Roles and Responsibilities must be defined before the recovery process.

Order of Recovery
The order and priority of the data and systems to be restored must be established prior to the recovery process.

Documentation Recovery Procedures
Procedures outlining how to recover IT systems and IT services must be documented in a clear and precise way, including commands for specific tools. The documentation must be stored in way that it can be accessed in case of disaster, e.g., in a printed version.

Documentation of Recovery
The recovery from backups must be documented, including date, time, duration, source system, target system, status (success, error, ...)

Safe Recovery
The recovery must only be done from non-compromised backups.

Recovery Tests
The recovery of backed up data must be tested quarterly to ensure the data can be recovered successfully when needed. A report of the test results must be produced and reviewed, and must contain at least date, time, duration, source system, target system, status (success, error).

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: CON.3 Datensicherungskonzept

- *IT-Grundschutz Compendium*: DER.4 Notfallmanagement
- General Data Protection Regulation

4.7 Unified Communications and Collaboration Policy

Purpose

The purpose of this policy is to establish guidelines for secure communication and collaboration within *SCA Tool* by using Unified Communications and Collaboration (UCC) services to protect sensitive information, maintain data integrity, and ensure compliance with applicable regulations.

Scope

This policy applies to all employees, contractors, and third-party partners engaged in collaborative activities related to *SCA Tool*, regardless of the platform used for collaboration.

Definitions

Unified Communications and Collaboration (UCC) service
A Unified Communications and Collaboration (UCC) service integrates various communication tools such as voice, video, chat, and presence indicators into a single application. It facilitates collaboration through features like screen sharing, team chat areas, and shared data storage, often provided as a cloud service with capabilities for additional applications through open interfaces.

Policy Statement

Following criteria related to UCC services must be met:

Compliance
Only UCC services must be used which respect the GDPR.

Accessibility
Every user of a UCC service must be able to access it from at least one of their devices.

Availability
The availability of UCC services must be granted at all times if possible. Planned outages must be communicated to users with reasonable advance notice. A minimum of one alternative means of communication must be available to ensure the communication during downtime of a UCC service. At least one alternative communication channel must be logically and phys-

ically separated to assure communication and collaboration during a failure of the IT-infrastructure.

All users of UCC services must adhere to the following principles when collaborating:

Data Protection
Sensitive information must be encrypted before sharing, and only shared through approved channels or via direct messages. The credentials to decrypt sensitive information must not be shared via the same UCC service. During screen sharing, users must be aware not to share sensitive data or Personally Identifiable Information (PII). When using voice chat, users must take care not to transmit the conversations of other people who are not involved.

Access Control
Access to the UCC services must only be allowed to authorized users by *SCA Tool* personnel. Users must have only the necessary permissions to perform their tasks. Access to sensitive data must be based on the principle of least privilege.

Authentication
Strong, multi-factor authentication methods must be used for accessing UCC services.

Incident Reporting
Any security incidents or suspected breaches must be reported immediately to the designated security officer.

Training
All users must undergo regular training on secure collaboration practices and data security principles.

Enforcement

Failure to comply with this policy may result in disciplinary actions, up to and including termination of collaboration and/or employment.

Review and Revision

This policy is reviewed annually and may be revised at any time.

Effective Date

This policy is effective from April 1st, 2025.

Approval

This policy is approved by the Head of *SCA Tool*.

References

This policy has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: APP.5.4 Unified Communications und Collaboration (UCC)
- General Data Protection Regulation (GDPR)

4.8 Security and Compliance Audit Process

Purpose

This document presents steps for executing security and compliance audits of *SCA Tool*, in order to facilitate an efficient audit process, improve the communication between auditors and the development team, and strengthen their overall security posture.

Process

The subsequent steps offer an organized approach to preparing, executing, and following up on security and compliance audits:

Pre-Audit Contractual Agreements
Before the audit commences, the auditor should enter into a contract that clearly outlines the audit's scope, timeline, and deliverables. This contract must incorporate a Non-Disclosure Agreement (NDA) to protect sensitive information.

Scope and Methodology
The contract must specify the types of penetration tests that are permitted or prohibited, including distinctions between non-intrusive and intrusive tests. It should also define the level of documentation required at each stage.

Access and Resources
The auditor is to be provided with a comprehensive list of all services and applications to be audited, alongside access credentials and relevant documentation. Necessary access to the IT infrastructure is granted, with a clearly defined role (e.g., `security_auditor`), to which the single permissions are added (e.g., `pve_read`, `sentry_read`). This role is then assigned to the auditor's user account. The account is to be deactivated directly after the audit is concluded.

Permissions Management
If possible, these roles should be configured to allow only read-access to minimize risk during the audit. The principles of least privilege and need-to-know are to be implemented to ensure security while granting access.

Communication Channels
The auditor should have a clear line of communication with the technical teams and a list of responsible developers for each service. This enables prompt engagement and support when needed.

Awareness of Existing Issues
Before the audit begins, the auditor should be informed about any known vulnerabilities in the scope of the audit. This allows them to adjust their focus and prioritize issues effectively, enabling a more efficient allocation of development resources.

Testing Environment
Penetration testing is conducted on a copy of the production environment or a staging environment that closely mirrors production, to allow for accurate assessment without disrupting live services.

Regular Updates
Throughout the audit, the auditor should maintain regular contact with a designated supervisor to report critical issues that require immediate attention and discuss any emerging issues that warrant further investigation.

Integration with Development
The auditor is to be treated as an internal member of *SCA Tool*, and should be invited to developer meetings. This inclusion allows the auditor to provide insights on secure coding practices and compliance-related features during the development phase.

Vulnerability Reporting and Management
Discovered vulnerabilities with high severity must be communicated immediately to developers through a ticketing system to facilitate timely remediation.

Post-Audit Review
A post-audit review meeting should be conducted to discuss findings, lessons learned, and recommendations for future audits.

Continuous Improvement
The feedback from the audit process should be used to continually improve security practices and audit procedures.

Training and Awareness
Integrating findings after their mitigation into training programs for developers should be considered to enhance their understanding of security vulnerabilities and best practices in secure coding.

References

This process has been created on the basis of the following documents and regulations:

- *IT-Grundschutz Compendium*: DER.3.1 Audits und Revisionen

5 Mitigation Strategies

The following sections present possible mitigations of the findings from chapter 3. Each mitigation will begin with a reference to the corresponding finding. The mitigations are based on the requirements established by the policies outlined in chapter 4. Some mitigations involve source code adjustments for *SCA Tool Application*, while others encompass operational changes. All changes aim at improving the security of *SCA Tool*, *SCA Tool Application*, or both.

5.1 Usage of Outdated and Vulnerable Libraries

This (sub)section mitigates finding(s) of (sub)section ***3.1***.

Regular SAST will greatly increase the security of *SCA Tool Application*. In section 3.1 *Dependency-Track* was used, which must be hosted, configured, and maintained. Especially for a small project team, this can be too much overhead. As an alternative, *CVE Binary Tool*[1] by *Intel* can be used directly from the Command-Line Interface (CLI). Developers can run this tool on their machine out-of-the-box without creating an SBOM file first. *CVE Binary Tool* checks the following databases for CVE data:

- National Vulnerability Database
- Open Source Vulnerability Database
- Gitlab Advisory Database
- RedHat Security Database
- Curl Database

Additionally, a *GitHub Action* can be set up to run *CVE Binary Tool* after every commit to the `main` branch. The report can be saved in various formats, such as Hypertext Markup Language (HTML) and then be published via *GitHub Pages*[2] to be available for the whole team of *SCA Tool*. Alternatively, the official

[1] https://github.com/intel/cve-bin-tool

[2] https://pages.github.com/

CVE Binary Tool GitHub Action[3] can be added to a workflow of the *SCA Tool Application GitHub* project.

After acquiring knowledge about the existence of one or more vulnerabilities, there are multiple ways to proceed and mitigate the issue. First, every vulnerability can be checked if it is introduced in the *SCA Tool Application* source code and if so, how it puts *SCA Tool Application* at risk. The risk can be calculated with equation 3.1. After this, the risk can either be avoided, reduced, transferred or accepted. Preferably, the risk is to be avoided if possible. This approach brings an enormous overhead, because much analysis of the vulnerabilities is done, but it results mostly in one mitigation: updating the software packages. Especially in bigger projects with hundreds of outdated and vulnerable software components, this process is expensive.

Consequently, the following approach is recommended and should be implemented for addressing the vulnerable and outdated libraries identified in section 3.1: Every vulnerable package should be checked in its newest version for breaking changes between the currently used version and the newest one. If no breaking changes exist or the breaking changes do not affect *SCA Tool Application*, update the component to the newest version. If no new version is available for a component, the procedure described before should be applied: Checking if the vulnerability is applicable and if so, introduce a mitigation. Such mitigation strategies may include evaluating alternative libraries that offer equivalent functionality, disabling affected features that are not essential, or designing workarounds to circumvent direct usage of the vulnerable feature. Additionally, it may be beneficial to implement a different library that acts as a safeguard, effectively addressing the vulnerabilities present in the original library.

With automated E2E-tests, e.g., with *Playwright*[4] by *Microsoft*, the mitigation can be fast-forwarded by updating all packages without checking for breaking changes, and afterwards running the automated tests. Breaking changes should then be detected and the single responsible packages can be identified and looked into with more detail.

If a CVE is rated with multiple different scores and a mitigation already exists, an analysis, which score is the correct one, may be cost-intensive. Therefore, the highest score should be taken as reference and the mitigation should be prioritized and scheduled accordingly.

The outlined procedure may hold true for *SCA Tool* and *SCA Tool Application*, because *SCA Tool* is a small team in control of the provided software service. For bigger organizations with a dedicated security department, and for software, which is running on customer's dedicated hardware in critical infrastructure, the

[3] https://github.com/intel/cve-bin-tool-action/

[4] https://playwright.dev/

process may involve more analysis of the single vulnerabilities and the associated cost/risk-factor of possible downtime.

5.2 Identity and Access Management

This (sub)section mitigates finding(s) of (sub)section ***3.2****.*

5.2.1 Secure Password Policy

This (sub)section mitigates finding(s) of (sub)section ***3.2.1****.*

The following password policy should be applied to *Ory Kratos*. Since *Ory Kratos* does not restrict the complexity of a password, the required length must be at least `15` characters long (Temoshok, 2024, Chapter 3.1.1.2).

```yaml
selfservice:
  ...
  methods:
    password:
      enabled: true
      config:
        haveibeenpwned_enabled: true
        max_breaches: 0
        ignore_network_errors: false
        min_password_length: 15
        identifier_similarity_check_enabled: true
```

Listing 5.1: *kratos.yaml*: Secure Password Policy Values

5.2.2 Multi-Factor Authentication

This (sub)section mitigates finding(s) of (sub)section ***3.2.2****.*

In order to add MFA to *Ory Kratos* the settings in listing 5.2 must be applied. By setting `session.whoami.required_aal` to `highest_available`, the whole session is protected by MFA for users, who configured MFA.

The implementation of `webauthn` facilitates the use of physical second-factor authentication measures for users, including technologies such as FaceID[5] and YubiKey[6]. Configuring `totp` allows the user to utilize an *Authenticator App* that generates a Time-Based One-Time Password (TOTP)[7] every 30 seconds, while

[5] https://support.apple.com/en-us/102381

[6] https://www.yubico.com/products/how-the-yubikey-works/

[7] https://www.ory.sh/docs/kratos/mfa/totp

`lookup_secret` lets the user store *backup codes*[8] as backup if their *Authenticator App* is not available.

```
selfservice:
  ...
  session:
    whoami:
      required_aal: highest_available
  methods:
    webauthn:
      config:
        passwordless: false
        rp:
          display_name: SCA Tool Application
          id: scatool.com
          origin: https://app.scatool.com:4455
      enabled: true
    totp:
      config:
        issuer: app.scatool.com
        enabled: true
    lookup_secret:
      enabled: true
```

Listing 5.2: *kratos.yaml*: Adding MFA

5.2.3 Secure Logging Configuration

This (sub)section mitigates finding(s) of (sub)section ***3.2.3****.*

```
log:
  level: info
  format: json
  leak_sensitive_values: false
```

Listing 5.3: *kratos.yaml*: Deactivated Logging of Sensitive Values

Logging should be set up as shown in listing 5.3. By using `level: info`, only general operational events are logged, which reduces the size of generated log files. Events should be logged in a structured format, which allows for ingestion into log analysis tools for better monitoring and inspection. *Ory Kratos* supports structured JavaScript Object Notation (JSON). Sensitive data, such as PII, secrets, tokens, etc., should never be logged. `leak_sensitive_values: true` should only be set manually in a local development cluster for a specific use case, e.g., if an error was reported and it needs to be debugged. It should never be set in a configuration file, which is uploaded to a shared repository, to avoid unintentionally

[8] https://www.ory.sh/docs/kratos/mfa/lookup-secrets

running this configuration and therefore acting against the organization's data protection guidelines.

5.2.4 Secure Password Hashing

*This (sub)section mitigates finding(s) of (sub)section **3.2.4**.*

As stated in subsection 3.2.4, the current `cost` factor of `8` is too low. Hardening the current *bcrypt* configuration against modern CPUs can be done by setting the `cost` factor to `12`, which is the default value set by *Ory Kratos*. The resulting code snippet looks as in listing 5.4. This value will have to be adjusted upwards in the future, due to the advances in computational power and therefore needs reevaluation from time to time.

```
hashers:
  bcrypt:
    cost: 12
```

Listing 5.4: *kratos.yaml*: Adjusted `cost` Factor of *bcrypt*

While *bcrypt* is secure, OWASP recommends using *Argon2id* over *bcrypt*. *Argon2* won the Password Hashing Competition (PHC) in 2015, an open competition organized by experts of the security and cryptographic field, working at NIST, *Microsoft*, *FreeBSD* and *Johns Hopkins University*, among others (Aumasson, 2019). In contrast to *bcrypt* (Wiemer & Zimmermann, 2014), *Argon2* is memory-hard. This means, that the function yields for maximum memory filling rate. By this, offline password cracking attacks can be slowed down, since the amount of memory is limited and calculations cannot be parallelized indefinitely. Furthermore, *Argon2* makes effective use of multiple CPU-cores and is resilient against trade-off attacks, e.g., cracking with less memory in more time (Biryukov et al., 2017). The values set in listing 5.5 are example values taken from the documentation of *Ory Kratos*. By using the CLI tool shipped with *Ory Kratos* shown in listing 5.6, the ideal parameters can be determined according to the underlying hardware. Due to missing access to the cluster, these values are not listed in this Master's thesis.

```
hashers:
  argon2:
    parallelism: 1
    memory: 128MB
    iterations: 3
    salt_length: 16
    key_length: 32
```

Listing 5.5: *kratos.yaml*: Example of *Argon2id* Configuration

```
kratos hashers argon2 calibrate 1s
```

Listing 5.6: *Ory Kratos* CLI Tool Command for *Argon2id*

5.2.5 Forced Re-Authentication for Sensitive Features

*This (sub)section mitigates finding(s) of (sub)section **3.2.5**.*

The value for `privileged_session_max_age` can be used to define the duration after which a user must re-enter their password following a successful login in order to make changes to their profile. The official documentation[9] does not specify potential values; hence, various values must be tested by the developers. Ideally, a value of `0s` can be set. If this is not feasible, the value should be set to `1s`. By setting it to one second, a potential attacker who has stolen the session will find it nearly impossible to make changes to the profile, as even automated attacks require time to execute.

```
selfservice:
  flows:
    ...
    settings:
            ui_url: https://app.scatool.com/settings
            privileged_session_max_age: 1s
            required_aal: highest_available
    ...
```

Listing 5.7: *kratos.yaml*: Forcing Re-Authentication for Profile Changes

5.2.6 Single Active User Session

*This (sub)section mitigates finding(s) of (sub)section **3.2.6**.*

Rather than permitting multiple active sessions for each individual user, the application should restrict users to one single active session. Upon logging in with valid credentials, any prior active sessions should be invalidated, to ensure that only a single session is permitted. Additionally, if a user modifies their password, all active sessions should be terminated as well. This is important because the user may want to change their password due to a potential leak, thereby preventing unauthorized access to their account. The same goes for the account recovery workflow, during which the password must be changed. If the user logs out, they should be logged out of all of their devices. If the user wishes to continue working on another device, they can log in again on that device.

[9] https://www.ory.sh/docs/kratos/self-service/flows/user-settings#updating-privileged-fields

The hooks to be added to *kratos.yaml* based on the official documentation[10] are shown in listing 5.8.

```
selfservice:
   flows:
      ...
      settings:
         ...
         after:
            password:
               hooks:
                  - hook: revoke_active_sessions
      ...
      recovery:
         after:
            hooks:
               - hook: revoke_active_sessions
      ...
      logout:
      ...
         after:
            password:
               hooks:
                  - hook: revoke_active_sessions
            oidc:
               hooks:
                  - hook: revoke_active_sessions
      login:
      ...
         after:
            password:
               hooks:
               - hook: revoke_active_sessions
            oidc:
               hooks:
               - hook: revoke_active_sessions
      ...
```

Listing 5.8: *kratos.yaml*: Allow Only One Active Session per User

This policy effectively precludes the concurrent usage of a single account by multiple users, which is essential for ensuring accountability and traceability, and may also play a significant role in the development of future cost models, such as pricing tiers based on the number of users per organization.

[10] https://www.ory.sh/docs/kratos/session-management/revoke-sessions-hook

5.3 Secure Use of Credentials

*This (sub)section mitigates finding(s) of (sub)section **3.3**.*

All plain text credentials must be changed according to the password requirements stated in section 4.5. The credentials must not be stored in plain text again. Instead they have to be encrypted.

By mounting volumes with *Kubernetes Secrets*[11] as already done in *monolith* and *worker* templates, credentials can be inserted securely into the *SCA Tool Application's* source code. The official documentation[12] at kubernetes.io describes the process in detail. As no further information is available about the IT infrastructure and the integration process for sensitive content, no further *SCA Tool*-related advice can be given.

The created *Secret*-files must not be committed to *GitHub* in plain text. The credentials must be stored in a secure vault or password manager that is storing the passwords encrypted, and be included in the backup cycle. If possible, the credentials and their backups should be stored at a different location than the *SCA Tool Application's* source code. If the credentials must be shared across the team members, they must be shared following the criteria of section 4.5.

After changing all credentials and storing them encrypted, commits with the old plain text credentials should be removed from the commit history even if the associated risks seem low. If an adversary gains access to the repository, they may still be able to view the old credentials. This access could allow them to attempt to use these credentials with other services or glean information about the security practices of *SCA Tool*. Moreover, following best practices in security is crucial for maintaining the integrity of *SCA Tool* and their systems. Regularly cleaning up sensitive information from the repository is a good habit that reflects a commitment to security. This is particularly pertinent given the potential for reputational risk. If an adversary gets hold of the source code of *SCA Tool Application* and discovers plain text credentials, they could share them, leading to trust issues by users of *SCA Tool Application*. By proactively managing sensitive information, *SCA Tool* protects their reputation and demonstrates responsibility in handling data security. Additionally, the risk of accidentally using old credentials is significant. Future developers working on *SCA Tool Application* may unintentionally use the old credentials in different contexts or services, because they assume they are still valid. Since *SCA Tool* employs students, who do not have yet the profound knowledge and experience as fully trained developers, and may therefore more likely adopt techniques, which are already used in the project. This scenario could lead to security breaches or operational issues.

[11] https://kubernetes.io/docs/concepts/configuration/secret/

[12] https://kubernetes.io/docs/tasks/inject-data-application/distribute-credentials-secure/

5.4 Unique Credentials

*This (sub)section mitigates finding(s) of (sub)section **3.4**.*

All default credentials must be changed. Separate credentials must be used for each service and each stage. The requirements outlined in 4.5 must be adhered to.

5.5 Secure Implementation of API Key Generation

*This (sub)section mitigates finding(s) of (sub)section **3.5**.*

Since two analyses of the API key generation implementation were conducted, two mitigation strategies are documented in the following.

5.5.1 First Implementation

*This (sub)section mitigates finding(s) of (sub)section **3.5.1**.*

The API key must only be visible once to the user directly after they click the button `Create API-Key`. To decrease the risk of *shoulder surfing* attacks, the API key should be masked and only be readable after clicking a separate button.

The API key must have an expiration date. It safeguards against unauthorized access by ensuring that the API key does not remain valid indefinitely. Unauthorized access can happen if the API key gets leaked unknowingly to the user, but also if the API key was intentionally used in a *GitHub Action workflow*, which then got deactivated and then after some time got run again accidentally.

The API key should have an ID. The ID can be hidden to the user. The ID can then be used for communication between frontend and backend, e.g., when invalidating the API key. So instead of using the API key in the header of the `DELETE`-request, the ID can be used. Listing 5.9 shows the example from listing 3.13 using the exemplary ID *1234* instead of the API key itself.

```
DELETE /api/web/api-key/1234 HTTP/1.1
```

Listing 5.9: Syllabus of HTTP Request Header

After a reload of the site, it should not be possible to display the API key anymore. Furthermore, the date of creation and the expiry date should be displayed next to the API key entry.

If a user can create multiple API keys in the future, it is advisable to implement scopes for fine-granular access control. These scopes could then be set by the user during the creation of an API key and provide only access to features chosen by the user.

For better usability, the `Copy`-function should work.

5.5.2 Second Implementation

*This (sub)section mitigates finding(s) of (sub)section **3.5.2**.*

The input text field `API-Key Name` should be limited server-side to, e.g., 64 characters to avoid the misuse of database storage. For usability-reasons, the limitation should also be implemented client-side.

As outlined in subsection 3.5.2, the prefix `sca-<expiry-date>-tool` should be removed. Instead, a random application-wide secret used as a *pepper* should be utilized, as recommended in the *OWASP Password Storage Cheat Sheet*[13]. However, a *GitHub* issue[14] asking for support of *peppering* by *Ory Kratos*, was closed without implementation. A self-developed implementation by *SCA Tool* could be considered similar to the already implemented prefix, but with keeping the value secret and including it in the hashing process.

Depending on the use cases of the customers, it may be beneficial to limit the number of API keys per customer. This practice trains customers to invalidate unused API keys to make room for new keys that are actually needed. As a result, the security of the unused API keys is not compromised by the customer due to loss or insecure storage for short-term Proof of Concept (PoC) purposes. Furthermore, this approach can help to reduce the storage requirements of the database. Additionally, implementing a pricing model based on the defined number of API keys could be considered as a business strategy to cover the increased costs associated with the computing resources used.

5.6 Non-Production Deployments With HTTPS

*This (sub)section mitigates finding(s) of (sub)section **3.6**.*

The routes at `/scatool/k8s/charts/routes/templates/routes.yaml` must be adjusted to offer *SCA Tool Application* via HTTPS. Currently, they are set for HTTP only. The utilization of HTTPS and TLS with *Traefik* is shown in the official documentation[15].

The use of valid certificates and Domain Name System (DNS) would significantly facilitate the testing of security features. The risk of confusion that can

[13] https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html#peppering

[14] https://github.com/ory/kratos/issues/1379

[15] https://doc.traefik.io/traefik/https/overview/

arise when using IP addresses would be minimized through descriptive domain names, such as `app.scatool-test.com` or `staging.int.app.scatool.com`. Either a self-signed certificate or a certificate from the nonprofit Certificate Authority (CA) *Let's Encrypt*[16] may be utilized.

Certificates of *Let's Encrypt* are accepted by basically all clients out of the box. Since `testing` and `staging` are only accessible within the FAU VPN, they cannot respond to the *HTTP-01* challenges of the *Let's Encrypt* servers. Therefore, the *DNS-01* challenge, which requires domain names, must be used. When using *Cloudflare*, the official plugin *certbot-dns-cloudflare*[17] can be employed. The issuance and renewal of certificates should occur outside of the restarting *Kubernetes* cluster. Instead, the certificates should be integrated similarly to secrets through mounting. Otherwise, in the event of multiple faulty restarts and subsequent requests for certificate issuance, rate limiting by the *Let's Encrypt* servers may come into effect, potentially blocking issuance for several days.

To use self-signed certificates without error messages, a root CA must be created using *OpenSSL*[18], and its public key must be imported to all clients. The root CA must be renewed regularly based on the chosen expiration period. The new public key must then be imported to all clients. If the root CA is compromised, a new one must also be issued. Additionally, when using penetration testing tools, the public key may need to be provided for testing security features. This grants complete control over the certificates and avoids dependency on third parties.

Both options have their pros and cons, although a one-time configured automation for using *Let's Encrypt* certificates seems more desirable. This is particularly relevant in the university context, where developers of *SCA Tool*, such as students writing their thesis, change frequently, leading to repeated inquiries and issues arising from manual integration into their end devices or when they implement new server-side automations for *SCA Tool Application.*

5.7 Secure HTTP Header

*This (sub)section mitigates finding(s) of (sub)section **3.7**.*

Traefik can add, remove, and overwrite headers of its HTTP responses. As described in the documentation, for development purposes, the header `IsDevelopment` can be set to `true` to mitigate the side effects of certain headers during development.

[16] https://letsencrypt.org/

[17] https://certbot-dns-cloudflare.readthedocs.io/en/stable/

[18] https://openssl-library.org/

```yaml
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
  name: header
spec:
  headers:
    customRequestHeaders:
    customResponseHeaders:
      Server: ""
      X-Powered-By: ""
      X-AspNet-Version: ""
      X-AspNetMvc-Version: ""
    accessControlAllowCredentials: false
    accessControlAllowMethods:
      - "GET"
      - "POST"
    accessControlAllowOriginList:
      - "https://scatool.com"
      - "https://app.scatool.com"
    accessControlMaxAge: 5
    allowedHosts:
      - "scatool.com"
      - "app.scatool.com"
    hostsProxyHeaders:
      - "X-Forwarded-Hosts"
    sslProxyHeaders:
      X-Forwarded-Proto: "https"
    stsSeconds: 31536000
    stsIncludeSubdomains: true
    stsPreload: true
    forceSTSHeader: false
    frameDeny: true
    contentTypeNosniff: true
    browserXssFilter: true
    contentSecurityPolicy: "default-src ’self’, img-src ’self’"
    referrerPolicy: "same-origin"
    isDevelopment: false
```

Listing 5.10: *traefik.yaml*: Secure HTTP Headers

Some of the headers of listing 5.10 are listed as guidance to ship responses with secure headers and to avoid vulnerabilities against common attack scenarios, such as *Cross-Site-Scripting (XSS)*. Since these headers could not be tested, they may be somewhat too restrictive. Consequently, they need to be relaxed incrementally. It may be even the case, that some headers must be changed completely to work with *SCA Tool Application*. The *MDN Web Docs*[19] can serve as a comprehensive resource for documentation on HTTP headers, as well the

[19] https://developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Headers

OWASP HTTP Security Response Headers Cheat Sheet[20] with recommendations on secure HTTP headers. By using *HTTP Observatory*[21] or *Security Headers*[22], HTTP headers of publicly facing websites, such as `production`, can be tested for compliance with best web security practices.

HTTP headers sent with responses of the *SCA Tool Application's* API may differ from the headers of the web User Interface (UI). Following the *OWASP REST Security Cheat Sheet*[23] can greatly increase the security of the API.

5.8 Secure Protocols and Ciphers

*This (sub)section mitigates finding(s) of (sub)section **3.8**.*

The German open source project *ciphersuite.info*[24] aggregates data about cipher suites from *IANA*[25], *OpenSSL*[26] and *GnuTLS*[27]. With their website, cipher suites can be checked for their security.

For generating secure web server configurations, the *Mozilla SSL Configuration Generator*[28] should be used. The `Mozilla Configuration` should be set to `Modern`, which allows only *TLSv1.3*. Rescorla (2018) introduced *TLSv1.3* in 2018. Since then, support for *TLSv1.3* was implemented over time in all modern web browsers, which can be expected to be used by organizations being users of *SCA Tool Application*.

Under `Miscellaneous`, `HTTP Strict Transport Security` and `OCSP Stapling` should be checked, if supported by the `Server Software`. The website *caniuse*[29] can provide assistance to determine, which web technologies, such as *TLSv1.3*, are supported by which web browsers.

It is strongly advised to refrain from using the *SSL/TLS Server Config Generator*[30] available at ssl.org. This version is significantly outdated compared to the *Mozilla SSL Configuration Generator* and may pose security risks when generating secure and client-compatible configurations. This recommendation is particularly important as this generator may still appear in search engine results.

[20] https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Headers_Cheat_Sheet.html

[21] https://developer.mozilla.org/en-US/observatory

[22] https://securityheaders.com/

[23] https://cheatsheetseries.owasp.org/cheatsheets/REST_Security_Cheat_Sheet.html#security-headers

[24] https://ciphersuite.info/

[25] https://www.iana.org/

[26] https://www.openssl.org/

[27] https://www.gnutls.org/

[28] https://ssl-config.mozilla.org/

[29] https://caniuse.com/

[30] https://www.ssl.org/server-config-generator

5.9 Hiding Information About Infrastructure

*This (sub)section mitigates finding(s) of (sub)section **3.9**.*

While custom error pages are in use for *SCA Tool Application*, the *nginx* version should still be removed from the default error pages. As described in the *nginx* documentation[31], this is done directly in the *nginx* configuration as demonstrated in listing 5.11. In this process, the version number is removed from both the `Server`-header and the error page; however, the web server name continues to be displayed.

```
server {
   listen       80;
   server_name  frontend;
   client_max_body_size 100M;
   server_tokens off;

   location / {
   ...
   }
}
```

Listing 5.11: *nginx.conf*: Remove *nginx* Version

To completely remove the value of the `Server`-header in the response, it must be modified by a middleware in the upstream *Traefik*. As described in the official *Traefik* documentation[32], the `Server` header must be overwritten as shown in listing 5.12. Additional headers can be overwritten in case these headers may get utilized by the backend services in the future.

```
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
   name: remove-server-header
spec:
   headers:
      customResponseHeaders:
         Server: ""
         X-Powered-By: ""
         X-AspNet-Version: ""
         X-AspNetMvc-Version: ""
```

Listing 5.12: *middlewares.yaml*: Remove `Server`-Header

The removal of the version number is merely security through obscurity and only increases the resource expenditure for a potential attacker, without preventing

[31] https://nginx.org/en/docs/http/ngx_http_core_module.html#server_tokens

[32] https://doc.traefik.io/traefik/middlewares/http/headers/#adding-and-removing-headers

the success of an attack. It does not replace the regular application of security patches.

5.10 Rate Limiting

*This (sub)section mitigates finding(s) of (sub)section **3.10**.*

Ory Kratos does not yet support the limiting of repeated login requests in a specified time. An issue[33] at their *GitHub* project is open and actively worked on.

SCA Tool Application makes use of an *Traefik Application Proxy*[34]. This proxy allows setting a rate limit by configuring the `average`-rate and `burst`-rate. The `average`-rate specifies the maximum number of allowed requests from a single source within a specified time. The `burst`-rate on the other hand defines the maximum number of allowed requests across all sources. Traefik Labs (2022) states, that their implementation of rate limiting is based on the token bucket algorithm, which is illustrated by them as leaky bucket for multiple source IP addresses in figure 5.1.

[33] https://github.com/ory/kratos/issues/3037

[34] https://doc.traefik.io/traefik/

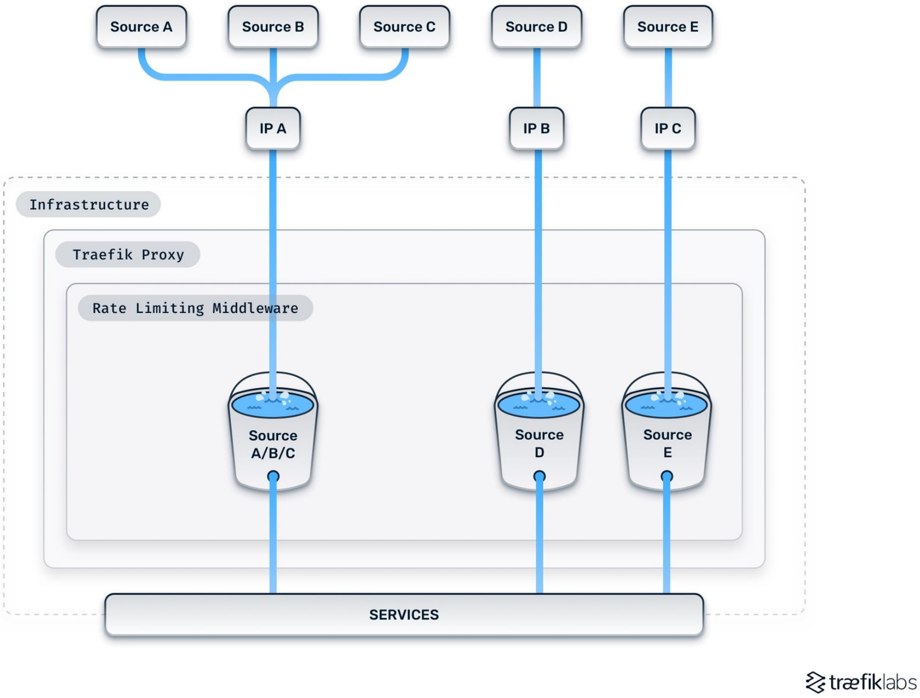


Figure 5.1: Visual Representation of the Token Bucket Algorithm (Traefik Labs, 2022)

Listing 5.13 shows a configuration for implementing rate limiting. In this example, the requests are grouped by the HTTP-header `identity`, which could be implemented to distinguish not by IP address but instead by username. This would effectively prevent password brute forcing for given usernames as well as misuse of *SCA Tool Application* after a hostile account takeover, even from multiple source IP addresses.

Additional source criteria can be found in *Traefik Labs*' documentation[35]. The values of `average` and `burst` should be roughly estimated based on the estimated number of requests per minute and user as well as on the specifications of the underlying IT infrastructure, and then tailored to the actual needs in order to fit it best without limiting valid requests. The values should be set more lax at first and then tightened based on monitored requests against the `production` deployment.

[35] https://doc.traefik.io/traefik/middlewares/http/ratelimit/

```
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
  name: ratelimit
spec:
  rateLimit:
    period: 1m
    average: 6
    burst: 120
    sourceCriterion:
    requestHeaderName: identity
```

Listing 5.13: *Traefik*: Rate Limiting Configuration Example

Rate limiting should also be activated on *Cloudflare*. This would move the workload of blocking requests from the *Traefik Proxy* to *Cloudflare*, while still having a second solution in place, if one of these two rate limiting solutions fail.

5.11 IP Allow List

*This (sub)section mitigates finding(s) of (sub)section **3.11**.*

To permit access to the deployments exclusively for *SCA Tool* members, an *IP Allow List* must be established in *Traefik*. This will filter requests based on the source IP address. Each member associated with the FAU who has access to the VPN is assigned a static IP address by the RRZE. This address can be viewed in the FAU IdMS at *https://www.idm.fau.de/go/profile/service/vpn.idms_username* (with *idms_username* to be replaced with the actual IdMS username) or in the network configuration settings of the member's end device when the VPN connection is established.

Listing 5.14 shows an example for the *Traefik* middleware configuration based on the documentation[36]. The entries `10.xxx.xxx.xxx/32` must be replaced with the *SCA Tool* members' static IP addresses. Due to missing access to the IT infrastructure, no proposition can be given about the use of `ipStrategy` and its possible `depth`-value.

[36] https://doc.traefik.io/traefik/middlewares/http/ipallowlist/

```
apiVersion: traefik.io/v1alpha1
kind: Middleware
metadata:
   name: ipallowlist
spec:
   ipAllowList:
   sourceRange:
      - 10.xxx.xxx.xx1/32
      - 10.xxx.xxx.xx2/32
      - 10.xxx.xxx.xx3/32
```

Listing 5.14: *Traefik* IP Allow List Example

While the *IP Allow List* reduces the attack surface of the deployments, it will not prevent *IP address spoofing*[37].

5.12 Backup Strategy

*This (sub)section mitigates finding(s) of (sub)section **3.12**.*

In order to be prepared for disaster recovery, the requirements stated in section 4.6 must be implemented. Data of *SCA Tool Application* and *SCA Tool* must be classified as critical, important or non-critical beforehand.

5.13 Storage Encryption on End Devices

*This (sub)section mitigates finding(s) of (sub)section **3.13**.*

New team members of *SCA Tool* must be instructed to download source code only to end devices with encrypted storage or otherwise protected against unauthorized access and theft. Given that some members of *SCA Tool* are utilizing devices that are not managed by the RRZE, it is obligatory that they undertake the responsibility of encrypting their devices on their own. The specific encryption process may differ based on the operating system employed.

For devices running *Windows*, the encryption can be accomplished using *BitLocker*. Comprehensive guidance on the use of *BitLocker* is available in the official documentation[38].

For *Linux* systems, storage encryption can be implemented through the use of *dm-crypt*. Detailed documentation regarding this process can be found in *Arch Linux's* wiki[39].

[37] https://www.cloudflare.com/learning/ddos/glossary/ip-spoofing/

[38] https://support.microsoft.com/en-us/windows/bitlocker-overview-44c0c61c-989d-4a69-8822-b95cd49b1bbf

[39] https://wiki.archlinux.org/title/Dm-crypt/Device_encryption

On *macOS*, it is necessary to utilize *FileVault* to secure data. Further information on how to protect data on *macOS* using *FileVault* is provided in the official guide[40].

Prior to the encryption of the drive, it is imperative to perform a system backup to ensure the ability to recover data in the event of an unforeseen issue or failure during the encryption process.

5.14 Security Through Local IT Service Providers

*This (sub)section mitigates finding(s) of (sub)section **3.14**.*

Due to existing issues and potential future problems with service providers outside the EU, particularly from the USA, comprehensive evaluations of these vendors are recommended. Given that *SCA Tool* is still relatively small and thus flexible, a timely transition would be easier than later when dependence on the utilized providers has increased due to growing infrastructure and customer base.

Should the decision be made to switch providers, websites such as *European Alternatives*[41] and *Go European*[42] can serve as a primary resource for identifying European alternatives to well-known vendors.

When selecting a provider, it is crucial to ensure that a DPA can be established with the vendor. To quickly identify providers that support this, the website *av-vertrag.org*[43] may be utilized.

For recommendations regarding IT security and data protection, the German IT security researcher and data protection advocate Mike Kuketz offers an recommendations section[44] on his website *Kuketz-Blog*[45]. This section includes not only IT service providers but also local software solutions. Among other resources, he provides a messenger matrix[46] that allows for the comparison of messaging services such as *Discord*.

It is important to note that independent research is still necessary, as the mentioned websites cannot guarantee completeness.

[40] https://support.apple.com/guide/mac-help/protect-data-on-your-mac-with-filevault-mh11785/mac

[41] https://european-alternatives.eu/

[42] https://www.goeuropean.org/

[43] https://av-vertrag.org/

[44] https://www.kuketz-blog.de/empfehlungsecke/

[45] https://www.kuketz-blog.de/

[46] https://www.messenger-matrix.de/messenger-matrix.html

5.14.1 Dismissal of Discord

*This (sub)section mitigates finding(s) of (sub)section **3.14.1**.*

As outlined in section 3.14.1, *Discord* must be replaced with a solution for team collaboration which respects IT security and data protection regulations. Due to the limited budget of the professorship, the new collaboration platform should be as inexpensive as possible while fulfilling the IT security and data protection requirements.

Based on the requirements, *Matrix*[47] in combination with the client *Element*[48] hosted by the RRZE is to be chosen.

The main benefits are the following. The user experience of *Element* is very similar to the one of *Discord*, which makes the migration smoothly and straightforward. Relevant statutory and legal provisions on data protection and IT security are complied with. The server location is in Erlangen, Germany. No private accounts of the *SCA Tool* members need to be used.

The open standard provides a federated real-time communication protocol that ensures global connectivity similar to email. It supports various forms of communication, including 1:1-chats and group discussions. The communication is decentralized, persistent, and interoperable without central control, allowing integration with other tools. It emphasizes data protection, featuring standard E2E-encryption for private conversations and group chats, with specific control over encryption settings at creation. The platform includes a web app, desktop client, and mobile apps, all developed independently. Users' mobile contacts remain private, as there is no automatic address book upload. It can integrate with existing authentication systems like the IdMS of the FAU. The platform is actively being developed and is gaining traction among other research institutions, including various German universities.

Access control, backup concept, etc., for the *Matrix* instance is implemented and maintained by the RRZE, a central facility of the FAU. Only members of the FAU can create spaces and rooms, but they can invite users with external *Matrix*-IDs.

The RRZE published a comprehensive guide[49] in German.

Along with the positive aspects, there are also associated risks. *SCA Tool Application*, *Matrix* and *FAUmail* are hosted on servers of the RRZE at the Erlangen site. If the server location fails, e.g., due to a cyber attack or a natural disaster, both the productive web application and the communication platform of the development team are no longer accessible.

[47] https://matrix.org/

[48] https://element.io/app-for-productivity

[49] https://www.anleitungen.rrze.fau.de/serverdienste/matrix-an-der-fau/

This risk can be mitigated by maintaining an additional communication method independent of the IT infrastructure of the RRZE. This can be achieved, for instance, through the installation of the instant messenger *Threema Work*[50]. According to Kuketz (2020), *Threema* is recommended, because the Swiss messenger prioritizes security and data protection, while a DPA can be concluded and no phone number for registration is required. This is particularly advantageous for small organizations, that do not have work mobile phones.

Alternatively, a list of phone numbers of *SCA Tool* members, with their consent, can be created for disaster recovery purposes. This list will then be securely and confidentially shared among the responsible individuals.

5.15 Email Notifications

*This (sub)section mitigates finding(s) of (sub)section **3.15**.*

Emails to customers should be sent from the official domain `scatool.com` or the subdomain `app.scatool.com` to establish trust with customers and strengthen the *SCA Tool* brand. Emails for development purposes from `testing` and `staging` should not be sent from Gmail addresses as well.

It may be beneficial to use separate domains instead of separate subdomains for `testing` and `staging`. This ensures that the environments are clearly separated and configurations between the various environments can be more easily copied or parameterized. Emails from the `testing` deployment could therefore be sent from, e.g., `noreply@scatool-test.com`.

Currently, *Gmail* is utilized as email server provider. Given the various legal retention periods for business emails, a transition to GDPR-compliant email providers within the European Union should be considered. For further details, please refer to section 5.14.

5.16 Measures Against User Lockout

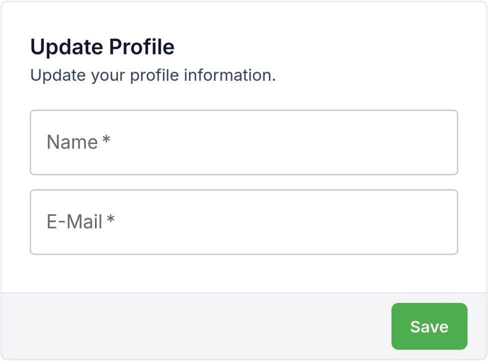
*This (sub)section mitigates finding(s) of (sub)section **3.16**.*

The current state of the `Update Profile`-dialog is seen in figure 5.2. No mechanism protects the user against accidental misspelling.

By adding a second mandatory input field to confirm the email address as seen in figure 5.3, this issue will be addressed. The new input field could be labeled with `Confirm E-Mail*` and must be required. It must not allow pasting into this field. The user must type their email address manually. If the values of `E-Mail*` and

[50] https://threema.ch/en/work/business-messenger

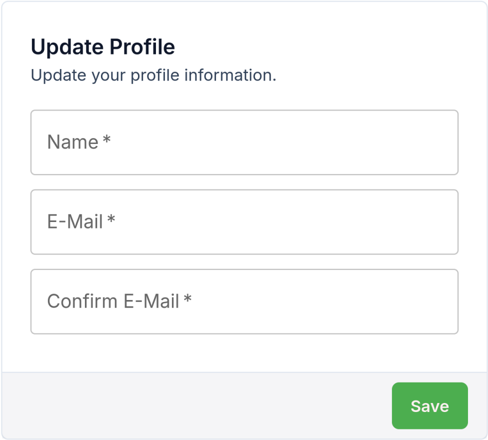
`Confirm E-Mail*` do not match, an error message, e.g., above the `Confirm E-Mail*`-field, must occur. The user must only be allowed to click the `Save`-button if both values match. The check if both values match can be implemented solely in the source code of the frontend. By implementing this feature, the user cannot lock themselves out by misspelling their email address.



Update Profile

Update your profile information.

Figure 5.2: `Update Profile`: Without E-Mail Confirmation Field



Update Profile

Update your profile information.

Figure 5.3: `Update Profile`: With E-Mail Confirmation Field

Additionally, the user should be informed via email to their old email address as soon as their email address was changed. If the action was not performed by the user, the user should be able to restore the old email address.

Every change in `Account` of the `User Settings` should be confirmed by MFA. MFA must not be done via email, but instead via *Authenticator App* or hardware token.

5.17 Measures Against Account Takeover

*This (sub)section mitigates finding(s) of (sub)section **3.16**.*

To mitigate Account Takeover (ATO) described in section 3.16, following countermeasures should be applied.

MFA must be implemented. MFA assures that if the first factor is compromised, the second factor is still protecting the user from their account being overtaken by an adversary. The second factor must be asked whenever the user logs into *SCA Tool Application*, every time when the user makes changes to their account settings, especially when they change their email address or their password, and also when the user requests a recovery code via the `Forgot your password?`-link. The second factor must not be implemented via email, since `Forgot your password?` is implemented via per email as well. Otherwise, an adversary would have access to the recovery password and the TOTP. Instead, the implementation of MFA should require an authenticator app or a hardware token, such as a *YubiKey*[51]. *Ory Kratos* supports MFA via *TOTP*, *WebAuthn*, *Lookup Secrets* and *Short Message Service (SMS)*. *Lookup Secrets* also known as *Backup Codes* or *Recovery Codes* must not be the only implementation of MFA due to their bad usability. *Lookup Secrets* should be shown to the user only once with an information text to store them in a secure place. *SMS* has been proven to be not secure enough (Jover, 2020) and create additional costs for *SCA Tool* (Karim et al., 2024). By this, *SMS* must not be implemented as MFA-method.

All active sessions of the user must be invalidated if the email address or the password was updated in *SCA Tool Application*. This ensures, that if the user suspects their credentials including the second factor being stolen, and therefore changes their credentials, an attacker, who is already logged into the user's account gets locked out immediately. This assumes, that the adversary has not altered the `Account`-settings yet. Furthermore, if the user logs out of *SCA Tool Application*, all active sessions must be voided. If an adversary obtained a user's session, e.g., if the user hasn't logged out of a shared device or the `ory_kratos_session`-cookie was stolen, they will be locked out of the user's account by this.

As already mentioned in section 5.16, the user should be informed via email to their old email address as soon as their email address was changed. If the action was not performed by the user, the user should be able to restore the old email address. If the notification includes the new email address, it should be masked to not hint the adversary.

5.18 Insufficient GitHub Access Control

*This (sub)section mitigates finding(s) of (sub)section **3.17**.*

GitHub provides countermeasures against unauthorized access. These are listed in their documentation[52].

To ensure protection against ATO, MFA must be set to be required. The IP address range of the FAU network must be added to the IP allow list. The ranges shown in listing 5.15 are documented on the website[53] of the RRZE and can change over time.

[51] https://www.yubico.com/products/

[52] https://docs.github.com/en/enterprise-cloud@latest/organizations/keeping-your-organization-secure

[53] https://www.rrze.fau.de/internet-e-mail/datennetz-der-fau/netzbereiche/

```
131.188.0.0/16
2001:638:a000::/48
```

Listing 5.15: IP Address Ranges of the FAU

Email notifications should be restricted to the domains `scatool.com`, `fau.de` and `group.riehle.org`.

6 Evaluation

In this evaluation, the results of this Master's thesis are compared to the requirements defined previously.

6.1 Reconsideration of Key Priorities

Initially, a closer alignment with the *ISO/IEC 27001* audit was planned. However, through a more in-depth engagement with the BSI *IT-Grundschutz*, it became evident that achieving a suitable level of protection according to the *IT-Grundschutz* also satisfies the requirements of *ISO/IEC 27001*. Given that the BSI provides the *IT-Grundschutz* resources and various additional materials free of charge and in an easily accessible manner, the focus shifted towards *IT-Grundschutz*. Since *SCA Tool* is still relatively young and its organizational structure develops gradually over time, a strict adherence to the *IT-Grundschutz* checklists did not seem practical. In this early stage, numerous changes of both technical and organizational nature are made, requiring continuous revision of established rules to adapt to new circumstances. The value of rules that cannot be adhered to is questionable and may hinder the healthy growth of the organization. Consequently, resources were allocated to tackle the most pertinent issues within *SCA Tool* that require urgent action to establish an organizational structure and ensure a secure *SCA Tool Application*. This foundational work can then serve as a basis for future developments, including the pursuit of certification under *IT-Grundschutz* or *ISO/IEC 27001*.

During the development of the policies, it became clear that it is difficult to formulate clear rules regarding the responsibilities and resources of *SCA Tool*. Due to insufficient insight into the organization concerning distributed tasks and resources, it was not possible to develop adequate processes. Therefore, it is advisable to conduct an inventory of activities, resources, and already existing processes at the beginning of the process creation. By using appropriate documentation through software, activities and resources can be related to each other, enabling the establishment of efficient processes. Stakeholders should be involved in the development of these processes.

Procedures that describe the execution of processes using suitable tools should be created by the individuals who regularly use these tools.

6.2 Implementation of Security and Data Protection

During the vulnerability assessment in chapter 3, weaknesses in various areas were identified. Vulnerabilities were discovered in parts of the application that directly communicate with the customer, such as the lack of MFA, insufficient password strength, during the creation of API keys, and the use of the domain *gmail.com* for sending communications. Other areas of concern include the hashing method of passwords and the repeated use of the same default credentials in plain text, which pertain solely to the source code. Insufficient data protection for customer and *SCA Tool* member data was revealed due to the use of insecure ciphers, reliance on service providers outside of the EU, and the absence of verification checks during profile data modifications.

The policies outlined in chapter 4 structured the areas of ISM, software development and operations, identity and access management, passwords, backup and recovery, as well as communication and collaboration, providing organizational solutions to the issues identified during the vulnerability assessment. The development of these policies incorporated information about the organization and their customer base.

Subsequently, specific operational solutions were proposed in chapter 5 to address the identified vulnerabilities in order to reduce the attack surface for potential threats and to strengthen *SCA Tool* organizationally.

6.3 Security Awareness

The necessary focus on information security of *SCA Tool* was merely given at the beginning of this Master's thesis. Instead, the primary objective was to implement non-security-relevant features. The simplest configuration gaps were present and a basic security standard did not yet exist as can be derived from the findings in chapter 3.

With the acceptance of this Master's thesis, the foundation was laid for a coordinated security process. Weekly meetings were held with the supervisor while working on this Master's thesis. During these meetings, the context of *SCA Tool* was discussed and security requirements for *SCA Tool* and *SCA Tool Application* were defined. They are reflected in the policies of chapter 4 and the security fixes in chapter 5.

Over the course of this thesis, the supervisor's security awareness increased as a result of these regular discussions. He transferred this awareness to his colleagues and other supervising students, which increased the overall security awareness throughout *SCA Tool*. He put the author of this Master's thesis in touch with two other of his supervising students to facilitate a discourse between them. On the one hand, general knowledge about information and IT security was brought into *SCA Tool*, and on the other hand, concrete implementation approaches for security-relevant features were evaluated. During these meetings, it became clear that there was an increased focus on the security of the web application and students were also encouraged to design *SCA Tool Application* to be secure.

Further student projects in the area of information security and IT security will be advertised in the future to ensure a sustainable security process. Some ideas based on this Master's thesis are introduced in chapter 8.

6.4 Introduction to Security Resources

An essential component, in addition to fostering security awareness among members of *SCA Tool*, is the provision of tools that assist in achieving the established security objectives.

In addition to self-created policies and code snippets as tools for the direct remediation of vulnerabilities, further online resources were presented. Comprehensive tools for both static and dynamic security application testing were introduced, as well as websites for information about secure configurations and online generators and checkers. Furthermore, numerous websites were mentioned that can assist in the selection of secure software solutions and services, while pointing out relevant legal considerations.

ISMS frameworks were also presented, which aid the organization in the structured attainment of information security through the establishment of an ISMS.

6.5 Organization of Security Audit

As this was the inaugural security audit for *SCA Tool*, the process did not proceed as smoothly as anticipated. There are several organizational aspects that could be enhanced to facilitate more effective execution of future audits.

Before the audit starts, the auditor should sign a contract which defines the time and the scope of the audit and includes an Non-Disclosure Agreement (NDA). The contract should state which types of penetration tests are allowed or forbidden, e.g., only non-intrusive tests, and the extent of documentation.

The auditor should be provided with a comprehensive list of all services to be audited, including information on how to access them. The auditor should be granted access to the IT infrastructure and its services. Specific roles should be created, e.g., `pve_auditor`, `nginx_auditor`, etc. and mapped to the auditor's user account. If possible, the roles should only allow read-operations. Additionally, the auditor should be given a list with all responsible developers of each service, to whom they can reach out in need of support.

Penetration testing should be done against a copy of the `production` deployment or a stage close to production, e.g., `staging`.

During the audit, the auditor should be in regular touch with the supervisor to inform them about critical issues, which need to be fixed as soon as possible, as well as to speak about arising security issues which should be investigated in more detail.

The auditor should be seen as an internal auditor. This means, that they are member of *SCA Tool* and that they should be invited to developer meetings so they can advise on correct implementation of security and compliance related features or bug fixes beforehand. Found vulnerabilities should be directly communicated to the developers, e.g., via ticket system. Already known issues should be communicated to the auditor, so the priority of topics can be shifted and they can organize their proceeding better. By this, development resources can be allocated more efficiently.

The regular at least weekly communication with the supervisor is worth to mention. This allowed for quick clarification of questions and discrepancies, the setting of priorities, and the acquisition of background information on specific issues. By invitation to *Discord*, rapid communication with other members of *SCA Tool* was also made possible. The communication with other *SCA Tool* members was actively promoted by the supervisor, enabling both sides to benefit from each other and for the greater good of *SCA Tool*.

Based on these lessons learned, the process in section 4.8 was created to ensure future efficient and effective security and compliance audits of *SCA Tool*.

7 Conclusions

In this chapter, conclusions are drawn about the results of the *SCA Tool* security audit.

7.1 Summary

A comprehensive security audit of *SCA Tool Application*, its IT infrastructure's configuration, and the information security management within the *SCA Tool* project team was conducted. The objective was to establish a foundation for an ISMS and to provide actionable recommendations for specific vulnerabilities. In the beginning it was comprehensively informed about ISMS frameworks and the significance of security audits. During the vulnerability assessment, software tools were utilized for penetration testing against *SCA Tool Application*, static source code analyses were conducted, and organizational and regulatory vulnerabilities were identified through questionnaires with the supervisor and meetings with developers. 18 primary issues were identified, while some of them included subordinate issues. Based on these findings and with guidance of existing ISMS frameworks, six policies and one process were developed for areas of *SCA Tool* where urgent action is required. Mitigation strategies for the previously identified vulnerabilities were formulated based on these policies. They included In this process, technical documentation of the software modules in use, as well as regulatory requirements and the organizational context, were taken into account. The mitigation strategies include secure configurations, compliance considerations, and additional resources for further information, such as links to online-generators and checkers to apply current best security practices. This thesis has demonstrated the urgent need for action regarding security in *SCA Tool* and *SCA Tool Application*. The importance of best practices in secure software development and consistency in source code has been highlighted. On top of that, it has been demonstrated that multiple layers of security are essential for a secure web application, as otherwise damage to the information of customers and the reputation of *SCA Tool* can be done. Finally, additional recommendations for the future enhancement of security for *SCA Tool* are provided in chapter 8.

7.2 Estimated Roadmap of Implementation

In order to achieve a timely release of *SCA Tool Application*, the following roadmap to mitigate issues found in chapter 3 is estimated based on their expected risk and time investment. This assessment is non-binding. The individual findings may be re-evaluated at any time, but should be addressed no later than one year.

The findings 3.1, 3.2, 3.7, 3.8, and 3.12 must be fixed until the next version of *SCA Tool Application* goes into `production` and therefore must be remediated in the short term.

The findings 3.3, 3.4, 3.5, 3.11, 3.14.1 and 3.14.2, and parts of 3.16 must be fixed in the mid term, saying within 3 to 6 months after the review of this Master's thesis.

The findings 3.6, 3.9, 3.10, 3.13, 3.14.3, 3.15, and 3.17 are not negligible and must be fixed in the long term, saying within one year after the review of this Master's thesis.

7.3 Promoting Best Practices in Secure Software Development

The findings in chapter 3 and their remediations in chapter 5 showed, that many issues arose from not following the official documentation of software solutions, such as *Ory Kratos* or *Traefik*.

It is important, that developers, who implement features in a security context, do not take shortcuts but instead take their time to configure the services thoroughly. For a quick PoC, shortcuts are fine, but they should not be committed to the *main* branch. The management of *SCA Tool* is required to support the developers in this approach and to put more emphasis on the security of *SCA Tool Application* instead of on the application's enrichment by new features. Developers should be motivated to use best practices regarding secure software development, e.g., by providing access to information, not blocking security recommendations, allowing participation in seminars, encouraging common exchange and mutual assistance, etc. The raised security awareness within *SCA Tool* during this Master's thesis and the shown interest by members of other projects of the professorship showed, that there is an intrinsic motivation for secure implementations, which can and should be used.

It can be beneficial to reference security-relevant documentation in feature tickets to assist with the implementation.

7.4 Consistency of Source Code

During the audit, it occurred, that the source code of *SCA Tool Application* included security features of the same technology in configuration files as well as in executing source code. The resulting lack of documentation, coupled with the growing volume of source code, may sooner or later lead to security-critical misconfigurations. Additionally, the deployments are called `testing`, `staging` and `production`, but the source code uses `dev`, `staging` and `production`. This led to some confusion during the audit. Consistent terminology across technologies and systems facilitates both the development and the audit of the technology concerned. Utilization of domain names clearly indicating the stage instead of IP addresses supports this.

7.5 Importance of Multi-Layered Security

The findings presented in sections 3.2, 3.10, 3.12, 3.15 and 3.16 highlight the significance of individual security measures. A chain of events described in section 3.16 led to the formation of an attack that could have substantial repercussions for both the client and *SCA Tool*. Utilizing a legitimate email domain would have protected the user from falling victim to the phishing email. Additionally, preventing parallel sessions would have raised the user's awareness of potential threats due to the logout and may have invalidated the attacker's session by a direct subsequent login of the user. Implementing rate limiting could have impeded the automated exploitation of the user-invite function, while a properly configured monitoring system could have alerted *SCA Tool* to suspicious activities. Furthermore, re-authentication measures could have thwarted any changes to access credentials, thereby preventing complete account takeover. Although the aforementioned vulnerabilities exist, the implementation of MFA could have mitigated the risk of successful exploitation in the first place.

Additionally, the described incident underscores the necessity of a robust backup strategy, as it remains uncertain what actions the attacker may have undertaken with the user's account, and the user should not be burdened with the task of fully recreating a new account.

From the aforementioned considerations, it can be concluded that comprehensive security can only be achieved through multiple layers. These layers encompass application-level, infrastructure-level, and organizational-level security, and they can complement one another. Should one security measure fail, other measures must ensure safety or, at a minimum, reduce the risk to an acceptable level.

8 Future Work

The following outlines future initiatives and ideas that could not be implemented in this Master's thesis due to time constraints or because they were outside the scope of the project. The mentioned items aim to provide guidance for further improvements for *SCA Tool* and *SCA Tool Application* within the realms of information security, cybersecurity, and compliance.

8.1 Patching of Vulnerable Software Code

The vulnerabilities found and suggestions for remedying them were communicated to *SCA Tool*. These vulnerabilities will be rectified in a timely manner by the developers of *SCA Tool Application*. The developers are required to track the individual findings and their progress during the remediation process. The relevant documentation of the software used, e.g., of *Ory Kratos*, should be followed. This Master's thesis will only be published after the software- and infrastructure-side vulnerabilities have been fixed in order to offer any potential adversaries a reduced attack surface. The published version will be masked partly to protect the personal data of data subjects, which were part of the findings. Rules defined in the policies of chapter 4 can already be applied in the process. The vulnerabilities found should be completely eliminated within the next three months.

8.2 Automated Static and Dynamic Application Security Testing

As mentioned in section 5.1, an *GitHub Action* should be set up to run tools such as *cve-bin-tool* against the source code of *SCA Tool Application* every time a commit to the `main` branch is done. Alternatively, the SAST-tool could run once every night. The *GitHub Action cve-bin-tool-action*[1] can be integrated directly into the existing *SCA Tool Application GitHub* project making use of the

[1] https://github.com/intel/cve-bin-tool-action

`Security`-tab. More information about *CVE Binary Tool* in combination with *GitHub Actions* can be found in the official documentation[2].

In addition to the mentioned SAST-tool, a DAST-tool such as *ZAProxy*[3] can be integrated into the development process. Using DAST, security can be tested during the runtime of *SCA Tool Application*, which can lead to the detection of vulnerabilities, that are only present by the combination of multiple modules interacting with each other. To continuously ensure security, a *GitHub Action*[4] of *ZAProxy* can be established for this purpose.

8.3 Penetration Testing

After implementing the mitigations introduced in chapter 5, a whitebox penetration test should be performed. The aim of this penetration test is to identify technical vulnerabilities in *SCA Tool Application's* source code and infrastructure configurations. The BSI provides the documents *Ein Praxis-Leitfaden für IS-Penetrationstests* and *Durchführungskonzept für Penetrationstests* to offer guidance in the setup of such an penetration test.

8.4 Evaluation of IT Service Providers

Comprehensive evaluations of IT service providers are necessary. In particular, providers from outside the EU, such as *Cloudflare, Inc.*, *GitHub, Inc.*, and *Google*, should be thoroughly examined. However, German providers, such as the RRZE, may also warrant evaluation. These assessments should focus on cybersecurity, risk assessment, compliance with applicable data protection laws, and other legal requirements, as well as the compatibility of the new provider's products with the existing infrastructure of *SCA Tool*. Consideration should be given to both the current status of *SCA Tool* and its planned future growth. If a migration from providers outside the EU to providers within the EU is planned, the links provided in section 5.14 can offer initial guidance.

8.5 Detection and Monitoring

Sentry[5] was set up, which represents an initial step towards a centralized log aggregation solution. The next phase involves attempting to extend *Sentry's* func-

[2] https://cve-bin-tool.readthedocs.io/en/latest/README.html#using-cve-binary-tool-in-github-actions

[3] https://www.zaproxy.org/

[4] https://github.com/zaproxy/action-full-scan

[5] https://sentry.io/welcome/

tionality to utilize it as a SIEM tool. The challenge include the implementation of log forwarders that can transmit not only application logs but also additional data, such as server resource utilization and network logs, to the *Sentry* instance. Alternatively, other software solutions can be evaluated for their potential application in *SCA Tool*. Furthermore, custom heuristics can be developed to detect anomalies in log data and alert the relevant personnel accordingly. This alerting can be accomplished via email and through leveraging UCC platforms. Based on an analysis of the logged data, additional security measures can be recommended.

8.6 Incident Management

When an attack is detected and a potential compromise of the IT infrastructure is identified, emergency measures must be initiated. It is necessary to define the procedures to follow in an emergency situation, including who is responsible for what, who will be informed at which points in time, and how communication will be maintained despite the failure of the infrastructure.

8.7 Further Development of the ISMS

Through a comprehensive structure analysis and following protection requirements analysis according to *IT-Grundschutz*, requirements for *SCA Tool* can be identified. Utilizing the checklists presented in subsection 2.2.2, the ISMS of *SCA Tool* can be systematically developed and extended. As described in chapter 4, this process can not only lead to the development of a concept for the *SCA Tool*, but also to a cross-project concept for the *Professorship for Open-Source Software.*

8.8 Design of a Comprehensive Developer Training Program

A comprehensive training program can be developed, which will train inexperienced developers and reinforce the knowledge of experienced developers. This training can not only include the *OWASP Top Ten*[6] but also the *OWASP Cheat Sheet Series*[7], with a focus on the technologies and programming techniques utilized by *SCA Tool*. Furthermore, an onboarding process can be established, which will incorporate not only the aforementioned training materials but also the *IT-Grundschutz* training material[8] provided by the BSI.

[6] https://owasp.org/www-project-top-ten/

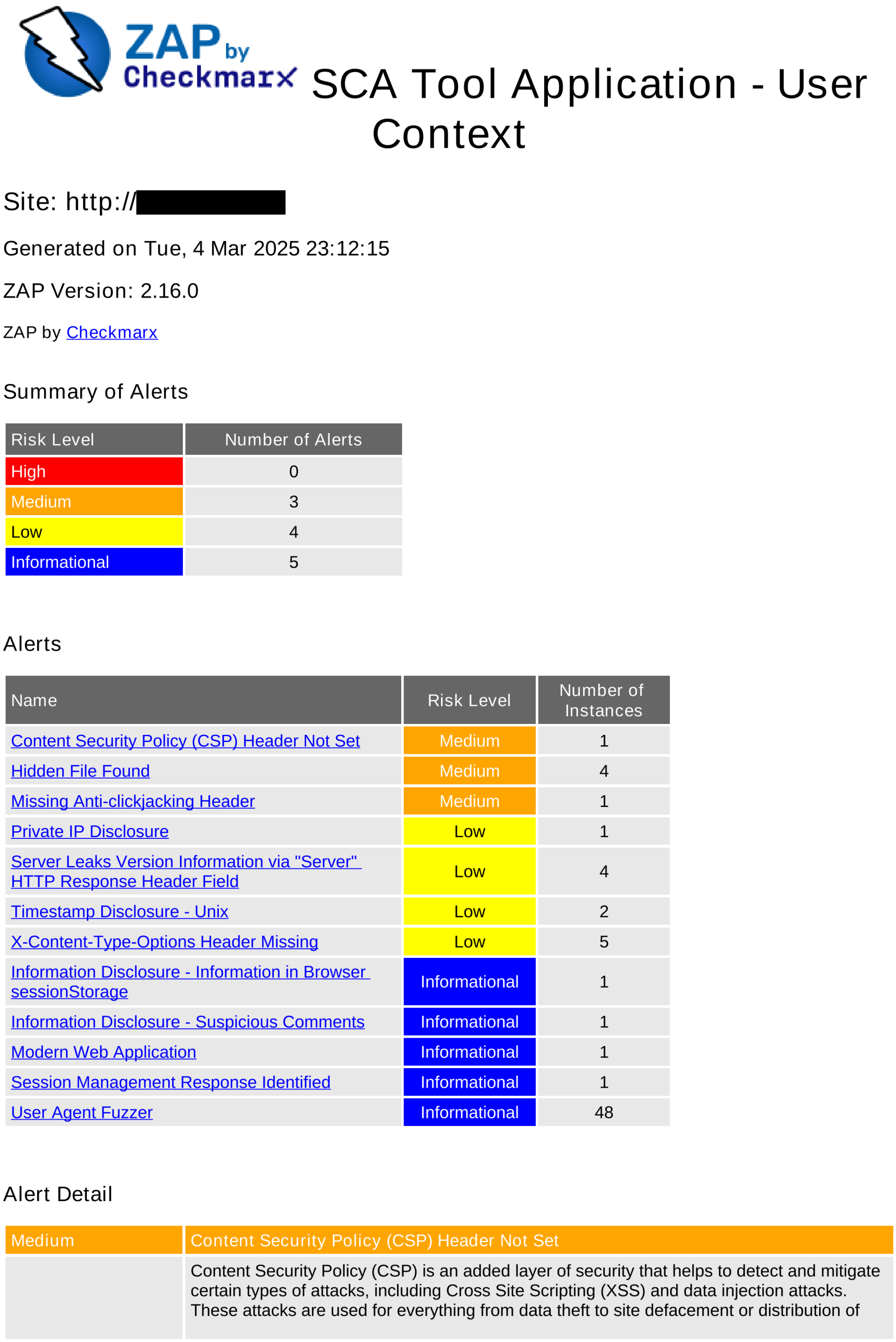
[7] https://cheatsheetseries.owasp.org/index.html

[8] https://www.bsi.bund.de/dok/10989992

Appendices

A *ZAP* Report of *SCA Tool Application* - User Context - REDACTED (testing)

The ZAP report of the test run against `testing` can be found on the following pages.



SCA Tool Application - User Context

Site: http://[redacted]

Generated on Tue, 4 Mar 2025 23:12:15

ZAP Version: 2.16.0

ZAP by Checkmarx

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	3
Low	4
Informational	5

Alerts

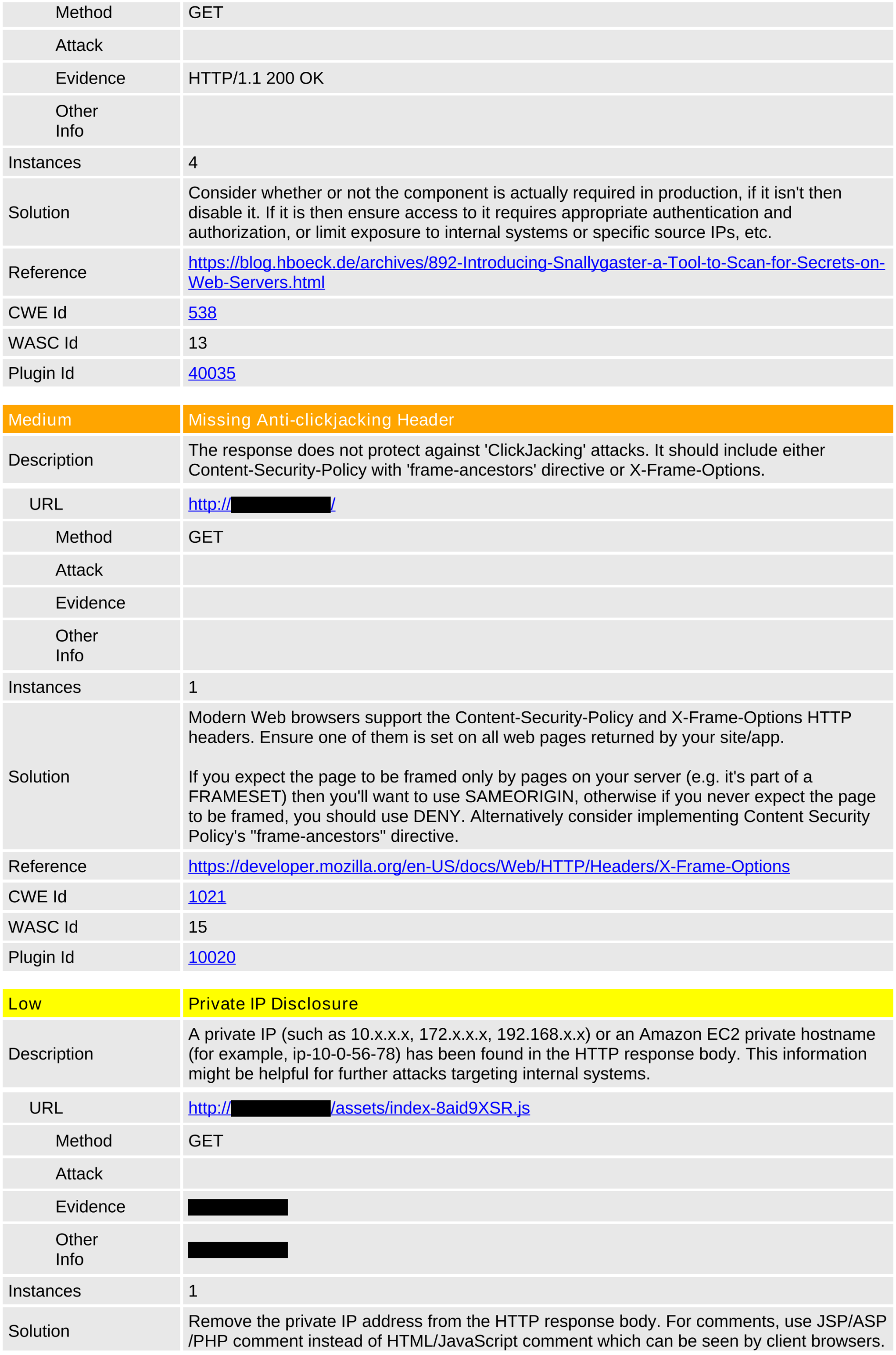
Name	Risk Level	Number of Instances
Content Security Policy (CSP) Header Not Set	Medium	1
Hidden File Found	Medium	4
Missing Anti-clickjacking Header	Medium	1
Private IP Disclosure	Low	1
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	4
Timestamp Disclosure - Unix	Low	2
X-Content-Type-Options Header Missing	Low	5
Information Disclosure - Information in Browser sessionStorage	Informational	1
Information Disclosure - Suspicious Comments	Informational	1
Modern Web Application	Informational	1
Session Management Response Identified	Informational	1
User Agent Fuzzer	Informational	48

Alert Detail

Medium	Content Security Policy (CSP) Header Not Set
	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of

Description	malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	
Other Info	
Instances	1
Solution	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.
Reference	https://developer.mozilla.org/en-US/docs/Web/Security/CSP/Introducing_Content_Security_Policy https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html https://www.w3.org/TR/CSP/ https://w3c.github.io/webappsec-csp/ https://web.dev/articles/csp https://caniuse.com/#feat=contentsecuritypolicy https://content-security-policy.com/
CWE Id	693
WASC Id	15
Plugin Id	10038

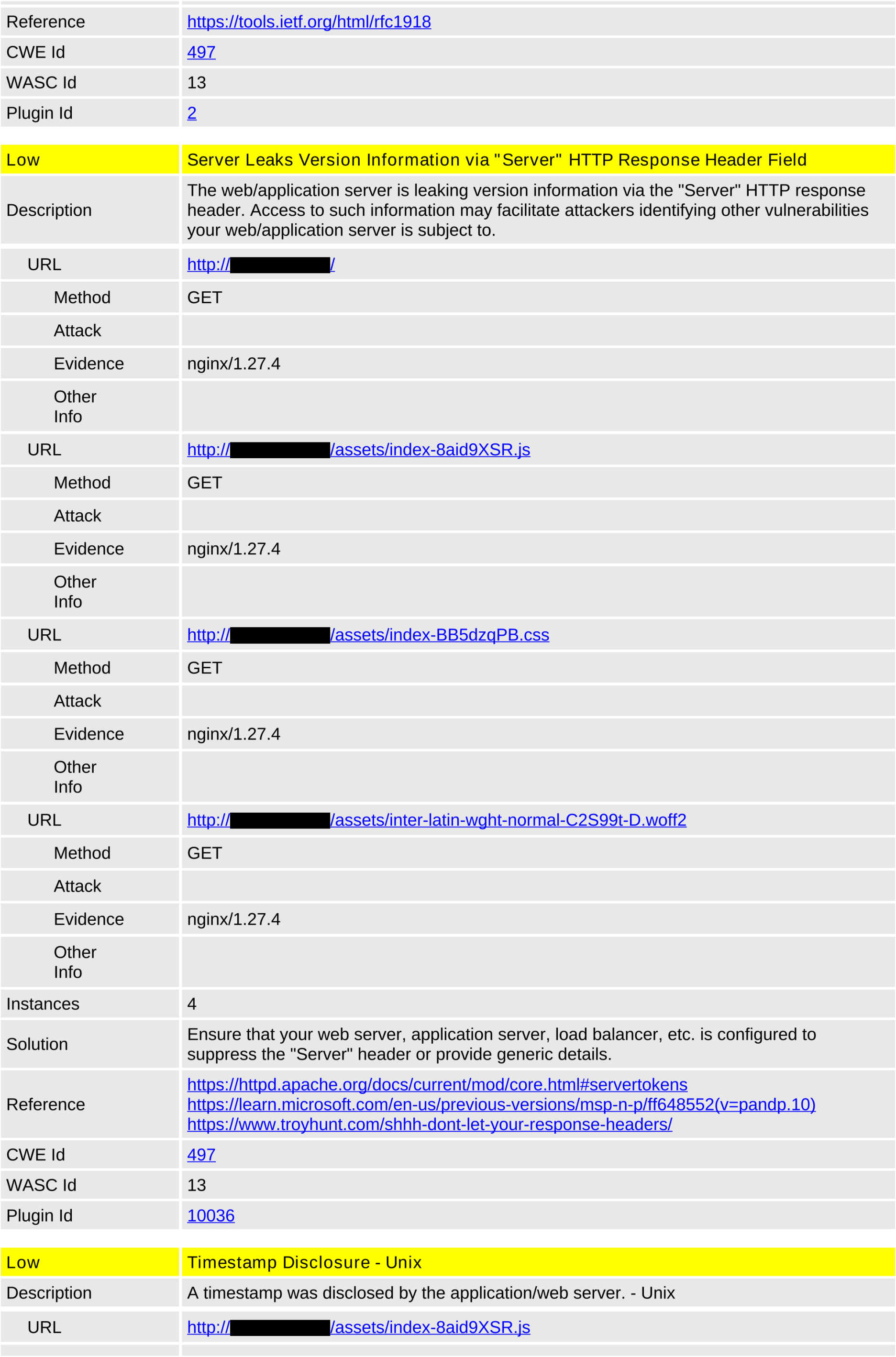
Medium	Hidden File Found
Description	A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.
URL	http://[redacted]/._darcs
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://[redacted]/.bzr
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://[redacted]/.hg
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://[redacted]/BitKeeper



Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
Instances	4
Solution	Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or specific source IPs, etc.
Reference	https://blog.hboeck.de/archives/892-Introducing-Snallygaster-a-Tool-to-Scan-for-Secrets-on-Web-Servers.html
CWE Id	538
WASC Id	13
Plugin Id	40035

Medium	Missing Anti-clickjacking Header
Description	The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	
Other Info	
Instances	1
Solution	Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app. If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.
Reference	https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
CWE Id	1021
WASC Id	15
Plugin Id	10020

Low	Private IP Disclosure
Description	A private IP (such as 10.x.x.x, 172.x.x.x, 192.168.x.x) or an Amazon EC2 private hostname (for example, ip-10-0-56-78) has been found in the HTTP response body. This information might be helpful for further attacks targeting internal systems.
URL	http://[redacted]/assets/index-8aid9XSR.js
Method	GET
Attack	
Evidence	[redacted]
Other Info	[redacted]
Instances	1
Solution	Remove the private IP address from the HTTP response body. For comments, use JSP/ASP /PHP comment instead of HTML/JavaScript comment which can be seen by client browsers.



Reference	https://tools.ietf.org/html/rfc1918
CWE Id	497
WASC Id	13
Plugin Id	2
Low	Server Leaks Version Information via "Server" HTTP Response Header Field
Description	The web/application server is leaking version information via the "Server" HTTP response header. Access to such information may facilitate attackers identifying other vulnerabilities your web/application server is subject to.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	nginx/1.27.4
Other Info	
URL	http://[redacted]/assets/index-8aid9XSR.js
Method	GET
Attack	
Evidence	nginx/1.27.4
Other Info	
URL	http://[redacted]/assets/index-BB5dzqPB.css
Method	GET
Attack	
Evidence	nginx/1.27.4
Other Info	
URL	http://[redacted]/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	nginx/1.27.4
Other Info	
Instances	4
Solution	Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.
Reference	https://httpd.apache.org/docs/current/mod/core.html#servertokens https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648552(v=pandp.10) https://www.troyhunt.com/shhh-dont-let-your-response-headers/
CWE Id	497
WASC Id	13
Plugin Id	10036
Low	Timestamp Disclosure - Unix
Description	A timestamp was disclosed by the application/web server. - Unix
URL	http://[redacted]/assets/index-8aid9XSR.js

Method	GET
Attack	
Evidence	1540483477
Other Info	1540483477, which evaluates to: 2018-10-25 18:04:37.
URL	http://[redacted]/assets/index-8aid9XSR.js
Method	GET
Attack	
Evidence	1831565813
Other Info	1831565813, which evaluates to: 2028-01-15 17:16:53.
Instances	2
Solution	Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.
Reference	https://cwe.mitre.org/data/definitions/200.html
CWE Id	497
WASC Id	13
Plugin Id	10096

Low	X-Content-Type-Options Header Missing
Description	The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://[redacted]/assets/index-8aid9XSR.js
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client

	or server error responses.
URL	http://[redacted]/assets/index-BB5dzqPB.css
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://[redacted]/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
Instances	5
Solution	Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages. If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application /web server to not perform MIME-sniffing.
Reference	https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/compatibility/gg622941(v=vs.85) https://owasp.org/www-community/Security_Headers
CWE Id	693
WASC Id	15
Plugin Id	10021
Informational	Information Disclosure - Information in Browser sessionStorage
Description	Information was stored in browser sessionStorage. This is not unusual or necessarily unsafe - this informational alert has been raised to help you get a better understanding of what this app is doing. For more details see the Client tabs - this information was set directly in the browser and will therefore not necessarily appear in this form in any HTTP(S) messages.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	
Other Info	The following data (key=value) was set: sentryReplaySession={"id":"3527f008fb5c4a6da261b9532a0e72e9","started":1741125809219,"lastActivity":1741125809219,"segmentId":0,"sampled":"session"} Note that this alert will only be raised once for each URL + key.
Instances	1
Solution	This is an informational alert and no action is necessary.
Reference	
CWE Id	359
WASC Id	13

Plugin Id	120000

Informational	Information Disclosure - Suspicious Comments
Description	The response appears to contain suspicious comments which may help an attacker.
URL	http://[redacted]/assets/index-8aid9XSR.js
Method	GET
Attack	
Evidence	Debug
Other Info	The following pattern was used: \bDEBUG\b and was detected in likely comment: "//${e.host}${n}${e.path?`/${e.path}`:""}/api/`}function IU(e){return`${EU(e)}${e.projectId}/envelope/`}function TU(e,t){const n=", see evidence field for the suspicious comment/snippet.
Instances	1
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Reference	
CWE Id	615
WASC Id	13
Plugin Id	10027

Informational	Modern Web Application
Description	The application appears to be a modern web application. If you need to explore it automatically then the Ajax Spider may well be more effective than the standard one.
URL	http://[redacted]/
Method	GET
Attack	
Evidence	<script type="module" crossorigin src="/assets/index-8aid9XSR.js"></script>
Other Info	No links have been found while there are scripts, which is an indication that this is a modern web application.
Instances	1
Solution	This is an informational alert and so no changes are required.
Reference	
CWE Id	
WASC Id	
Plugin Id	10109

Informational	Session Management Response Identified
Description	The given response has been identified as containing a session management token. The 'Other Info' field contains a set of header tokens that can be used in the Header Based Session Management Method. If the request is in a context which has a Session Management Method set to "Auto-Detect" then this rule will change the session management to use the tokens identified.
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	aehpVjk6vPeMPn2S2uCu1qtH0YGNclprZcRI1F5Ak98=
Other Info	cookie: csrf_token_6f7071aa25a34d3b8a81078e80c0b05ec3c0a5026c083fdea2642ba95bab611d
Instances	1

Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.
Reference	https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id
CWE Id	
WASC Id	
Plugin Id	10112

Informational	User Agent Fuzzer
Description	Check for differences in response based on fuzzed User Agent (eg. mobile sites, access as a Search Engine Crawler). Compares the response statuscode and the hashcode of the response body with the original response.
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36

Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	http://[redacted]/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	

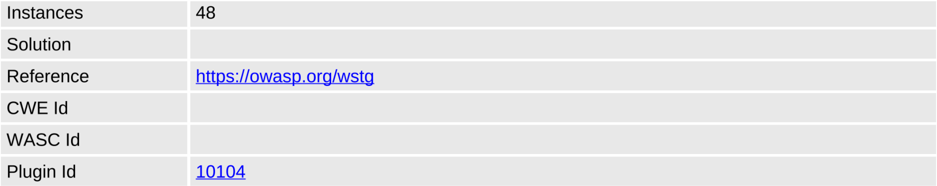
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	

Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	http://[redacted]/assets
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other	

Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other	

Info	
URL	http://[redacted]/images
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	http://[redacted]/images
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	

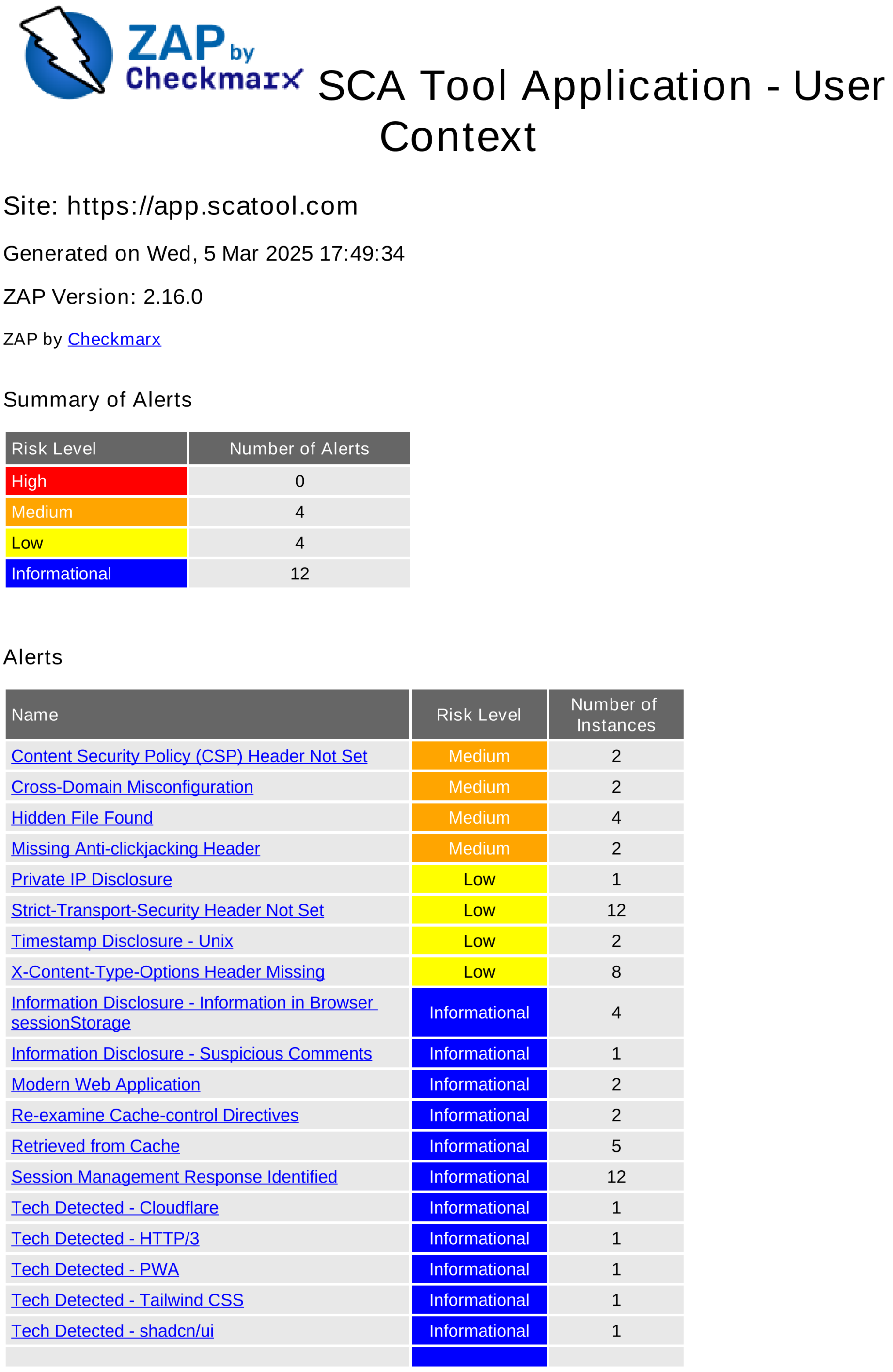
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	http://[redacted]/images/logos
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	



Instances	48
Solution	
Reference	https://owasp.org/wstg
CWE Id	
WASC Id	
Plugin Id	10104

B *ZAP* Report of *SCA Tool Application* - User Context - app.scatool.com (production)

The ZAP report of the test run against `production` can be found on the following pages.



SCA Tool Application - User Context

Site: https://app.scatool.com

Generated on Wed, 5 Mar 2025 17:49:34

ZAP Version: 2.16.0

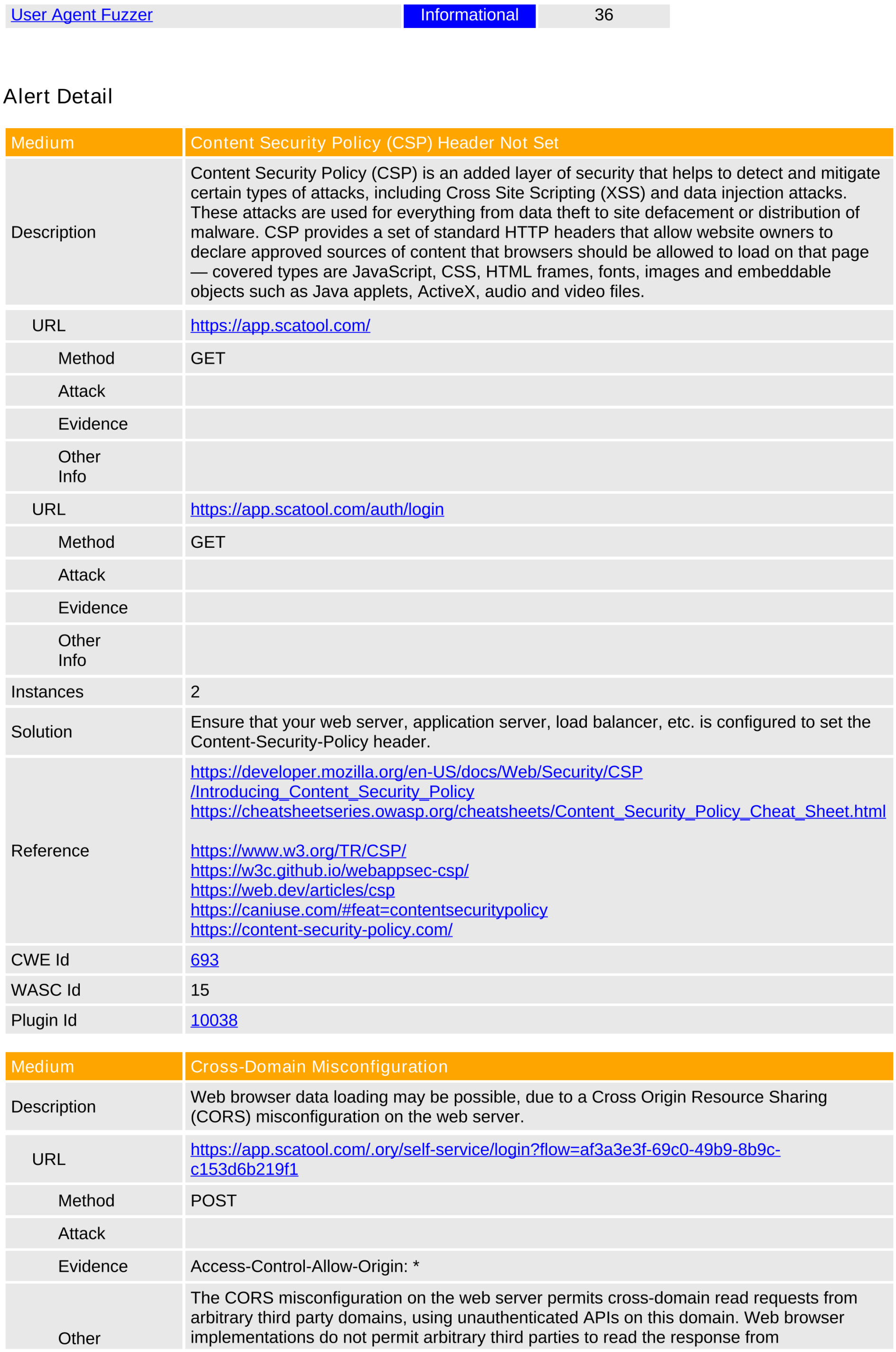
ZAP by Checkmarx

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	4
Low	4
Informational	12

Alerts

Name	Risk Level	Number of Instances
Content Security Policy (CSP) Header Not Set	Medium	2
Cross-Domain Misconfiguration	Medium	2
Hidden File Found	Medium	4
Missing Anti-clickjacking Header	Medium	2
Private IP Disclosure	Low	1
Strict-Transport-Security Header Not Set	Low	12
Timestamp Disclosure - Unix	Low	2
X-Content-Type-Options Header Missing	Low	8
Information Disclosure - Information in Browser sessionStorage	Informational	4
Information Disclosure - Suspicious Comments	Informational	1
Modern Web Application	Informational	2
Re-examine Cache-control Directives	Informational	2
Retrieved from Cache	Informational	5
Session Management Response Identified	Informational	12
Tech Detected - Cloudflare	Informational	1
Tech Detected - HTTP/3	Informational	1
Tech Detected - PWA	Informational	1
Tech Detected - Tailwind CSS	Informational	1
Tech Detected - shadcn/ui	Informational	1



User Agent Fuzzer	Informational	36

Alert Detail

Medium	Content Security Policy (CSP) Header Not Set
Description	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	
Instances	2
Solution	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.
Reference	https://developer.mozilla.org/en-US/docs/Web/Security/CSP/Introducing_Content_Security_Policy https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html https://www.w3.org/TR/CSP/ https://w3c.github.io/webappsec-csp/ https://web.dev/articles/csp https://caniuse.com/#feat=contentsecuritypolicy https://content-security-policy.com/
CWE Id	693
WASC Id	15
Plugin Id	10038

Medium	Cross-Domain Misconfiguration
Description	Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server.
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	POST
Attack	
Evidence	Access-Control-Allow-Origin: *
Other	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from

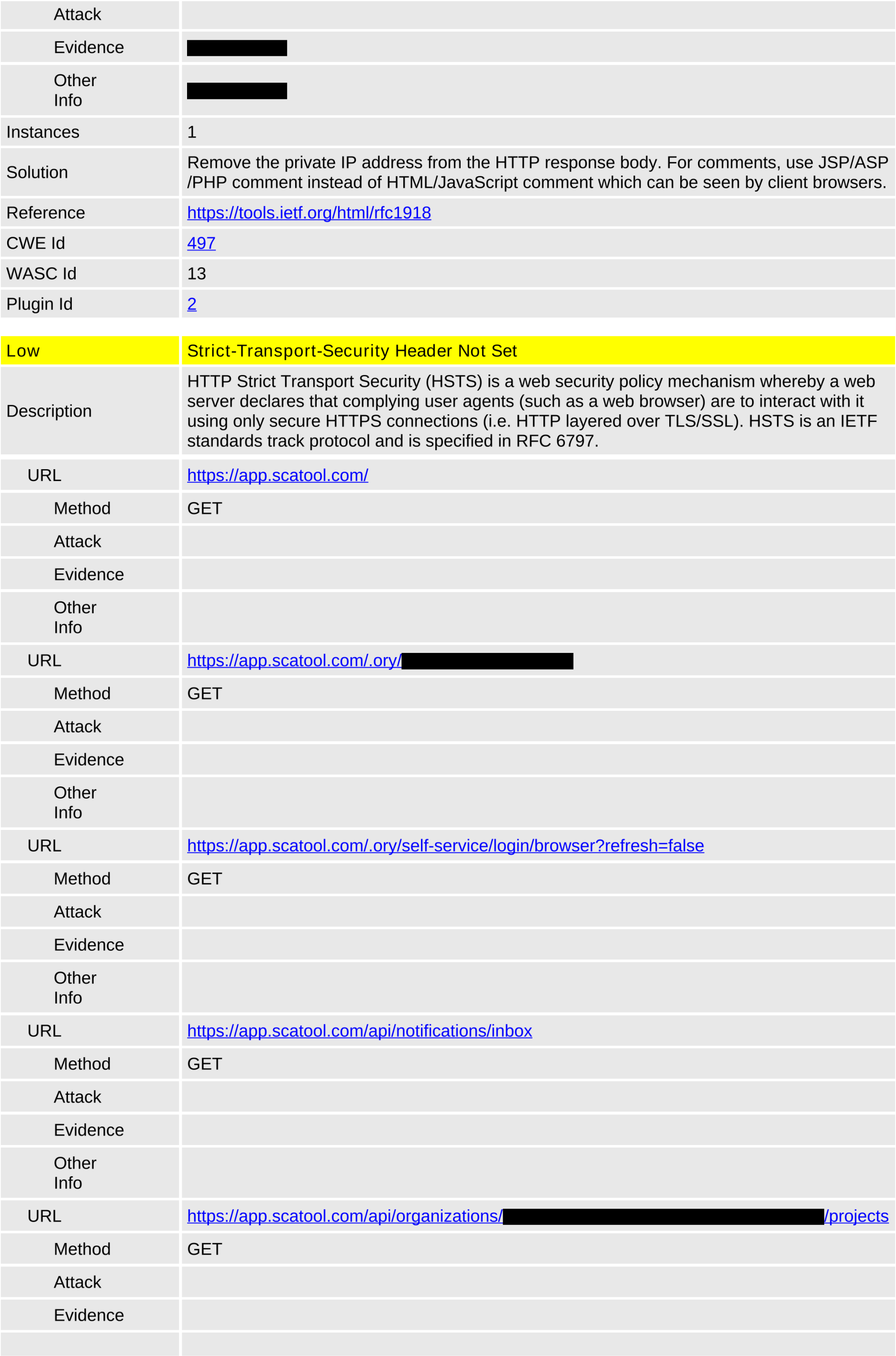
Info	authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
Instances	2
Solution	Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance). Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.
Reference	https://vulncat.fortify.com/en/detail?id=desc.config.dotnet.html5_overly_permissive_cors_policy
CWE Id	264
WASC Id	14
Plugin Id	10098

Medium	Hidden File Found
Description	A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.
URL	https://app.scatool.com/._darcs
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	https://app.scatool.com/.bzr
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	https://app.scatool.com/.hg
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	https://app.scatool.com/BitKeeper

Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
Instances	4
Solution	Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or specific source IPs, etc.
Reference	https://blog.hboeck.de/archives/892-Introducing-Snallygaster-a-Tool-to-Scan-for-Secrets-on-Web-Servers.html
CWE Id	538
WASC Id	13
Plugin Id	40035

Medium	Missing Anti-clickjacking Header
Description	The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	
Instances	2
Solution	Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app. If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.
Reference	https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
CWE Id	1021
WASC Id	15
Plugin Id	10020

Low	Private IP Disclosure
Description	A private IP (such as 10.x.x.x, 172.x.x.x, 192.168.x.x) or an Amazon EC2 private hostname (for example, ip-10-0-56-78) has been found in the HTTP response body. This information might be helpful for further attacks targeting internal systems.
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET



Attack	
Evidence	[redacted]
Other Info	[redacted]
Instances	1
Solution	Remove the private IP address from the HTTP response body. For comments, use JSP/ASP /PHP comment instead of HTML/JavaScript comment which can be seen by client browsers.
Reference	https://tools.ietf.org/html/rfc1918
CWE Id	497
WASC Id	13
Plugin Id	2

Low	Strict-Transport-Security Header Not Set
Description	HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/[redacted]
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	
Evidence	

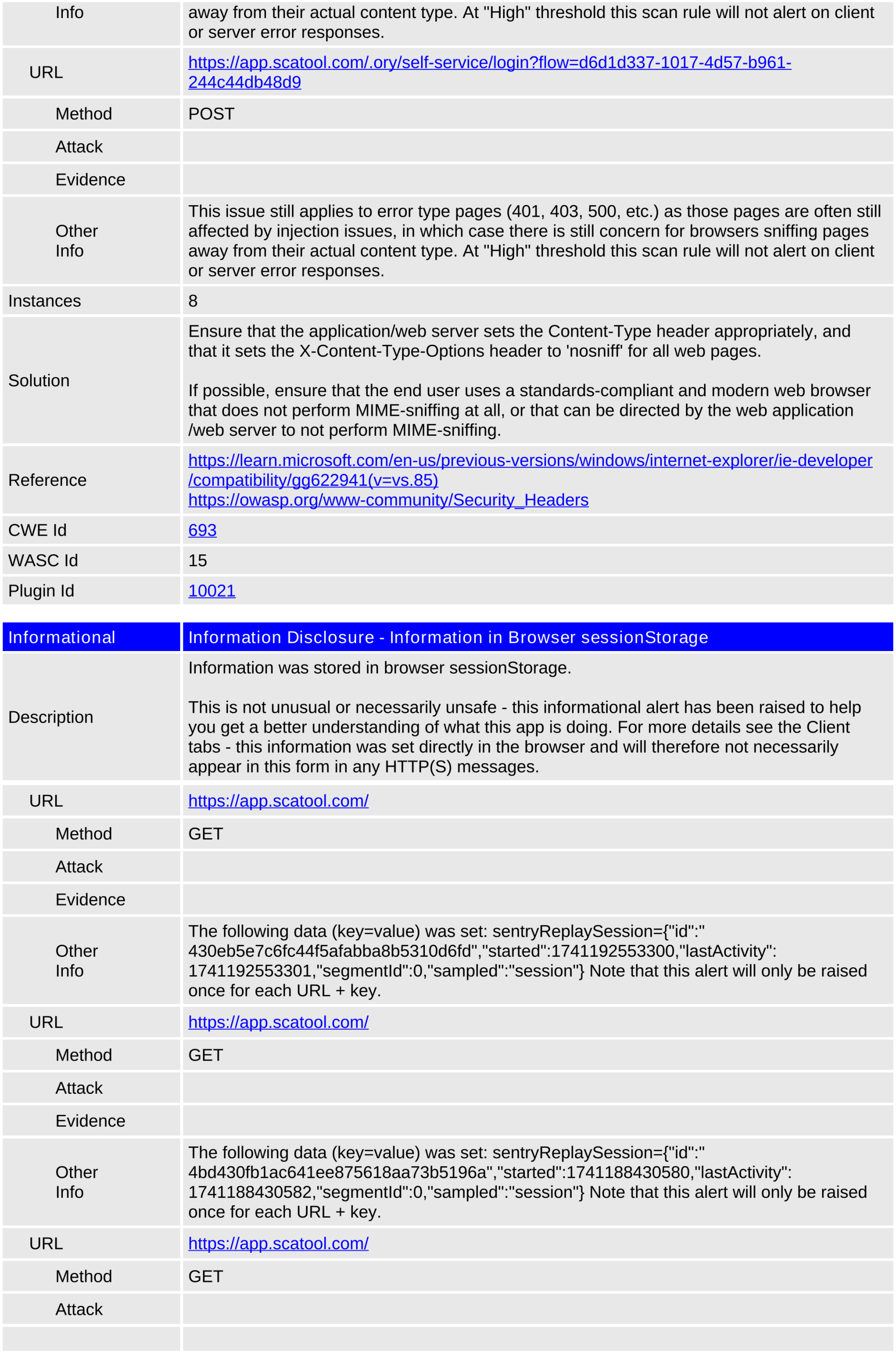
Other Info	
URL	https://app.scatool.com/api/users/user-info
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	POST
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	
Other	

Info	
Instances	12
Solution	Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.
Reference	https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html https://owasp.org/www-community/Security_Headers https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security https://caniuse.com/stricttransportsecurity https://datatracker.ietf.org/doc/html/rfc6797
CWE Id	319
WASC Id	15
Plugin Id	10035

Low	Timestamp Disclosure - Unix
Description	A timestamp was disclosed by the application/web server. - Unix
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	1540483477
Other Info	1540483477, which evaluates to: 2018-10-25 18:04:37.
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	1831565813
Other Info	1831565813, which evaluates to: 2028-01-15 17:16:53.
Instances	2
Solution	Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.
Reference	https://cwe.mitre.org/data/definitions/200.html
CWE Id	497
WASC Id	13
Plugin Id	10096

Low	X-Content-Type-Options Header Missing
Description	The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages

Info	away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	POST
Attack	
Evidence	
Other	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages



Info	away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
Instances	8
Solution	Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages. If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application /web server to not perform MIME-sniffing.
Reference	https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/compatibility/gg622941(v=vs.85) https://owasp.org/www-community/Security_Headers
CWE Id	693
WASC Id	15
Plugin Id	10021

Informational	Information Disclosure - Information in Browser sessionStorage
Description	Information was stored in browser sessionStorage. This is not unusual or necessarily unsafe - this informational alert has been raised to help you get a better understanding of what this app is doing. For more details see the Client tabs - this information was set directly in the browser and will therefore not necessarily appear in this form in any HTTP(S) messages.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	The following data (key=value) was set: sentryReplaySession={"id":"430eb5e7c6fc44f5afabba8b5310d6fd","started":1741192553300,"lastActivity":1741192553301,"segmentId":0,"sampled":"session"} Note that this alert will only be raised once for each URL + key.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	The following data (key=value) was set: sentryReplaySession={"id":"4bd430fb1ac641ee875618aa73b5196a","started":1741188430580,"lastActivity":1741188430582,"segmentId":0,"sampled":"session"} Note that this alert will only be raised once for each URL + key.
URL	https://app.scatool.com/
Method	GET
Attack	

Evidence	
Other Info	The following data (key=value) was set: sentryReplaySession={"id":" 6a04a7132efe4022993eecc60875be24","started":1741188586458,"lastActivity": 1741188586460,"segmentId":0,"sampled":"session"} Note that this alert will only be raised once for each URL + key.
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	The following data (key=value) was set: sentryReplaySession={"id":" 51dd47c04bee4d0ebe319a46c9d71281","started":1741192730006,"lastActivity": 1741192730000,"segmentId":0,"sampled":"buffer"} Note that this alert will only be raised once for each URL + key.
Instances	4
Solution	This is an informational alert and no action is necessary.
Reference	
CWE Id	359
WASC Id	13
Plugin Id	120000

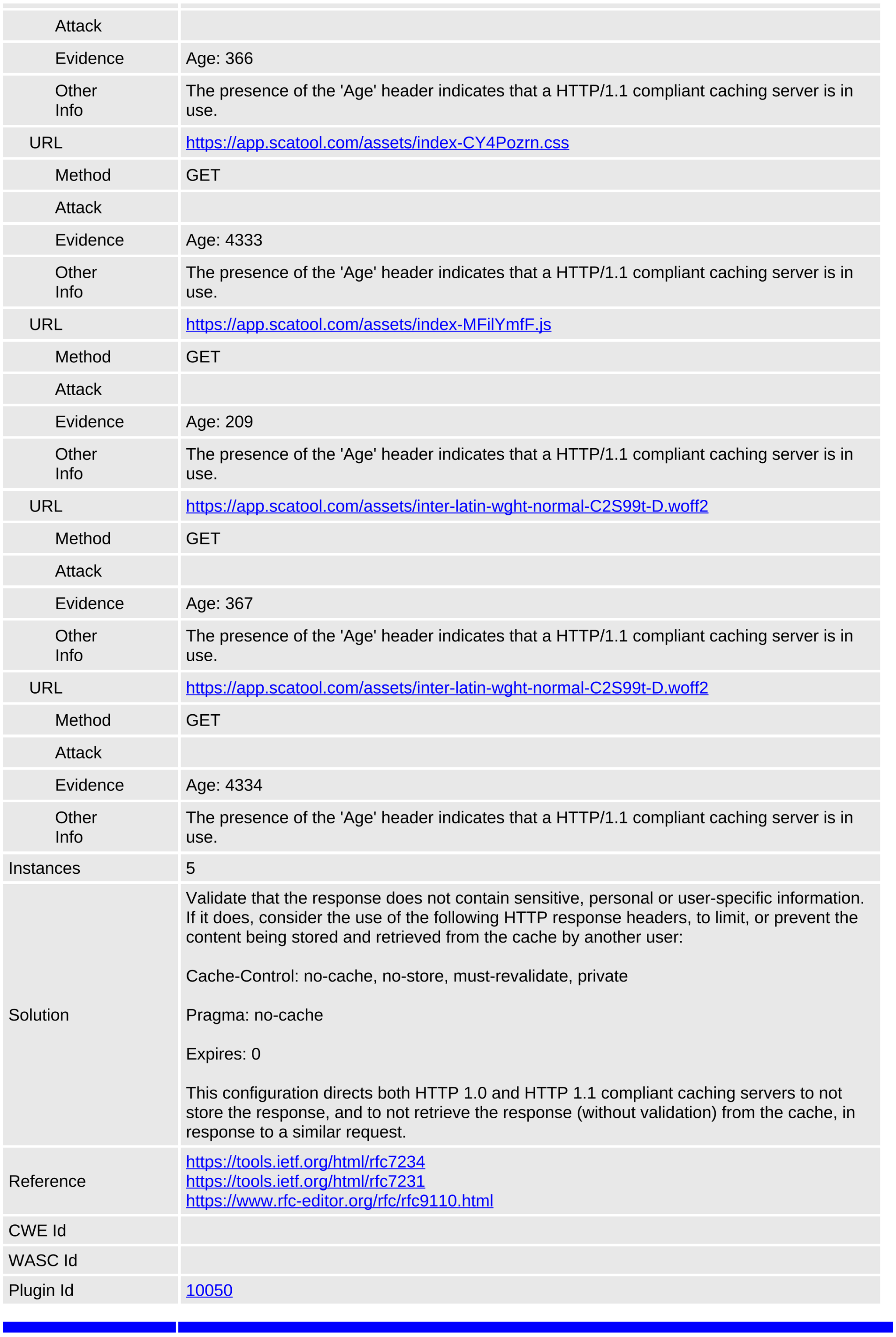
Informational	Information Disclosure - Suspicious Comments
Description	The response appears to contain suspicious comments which may help an attacker.
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	Debug
Other Info	The following pattern was used: \bDEBUG\b and was detected in likely comment: "//${e. host}${n}${e.path?`/${e.path}`:""}/api/`}function yU(e){return`${bU(e)}${e.projectId}/envelope /`}function vU(e,t){const n=", see evidence field for the suspicious comment/snippet.
Instances	1
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Reference	
CWE Id	615
WASC Id	13
Plugin Id	10027

Informational	Modern Web Application
Description	The application appears to be a modern web application. If you need to explore it automatically then the Ajax Spider may well be more effective than the standard one.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	<script type="module" crossorigin src="/assets/index-MFilYmfF.js"></script>
Other Info	No links have been found while there are scripts, which is an indication that this is a modern web application.
URL	https://app.scatool.com/auth/login
Method	GET

Attack	
Evidence	<script type="module" crossorigin src="/assets/index-MFilYmfF.js"></script>
Other Info	No links have been found while there are scripts, which is an indication that this is a modern web application.
Instances	2
Solution	This is an informational alert and so no changes are required.
Reference	
CWE Id	
WASC Id	
Plugin Id	10109

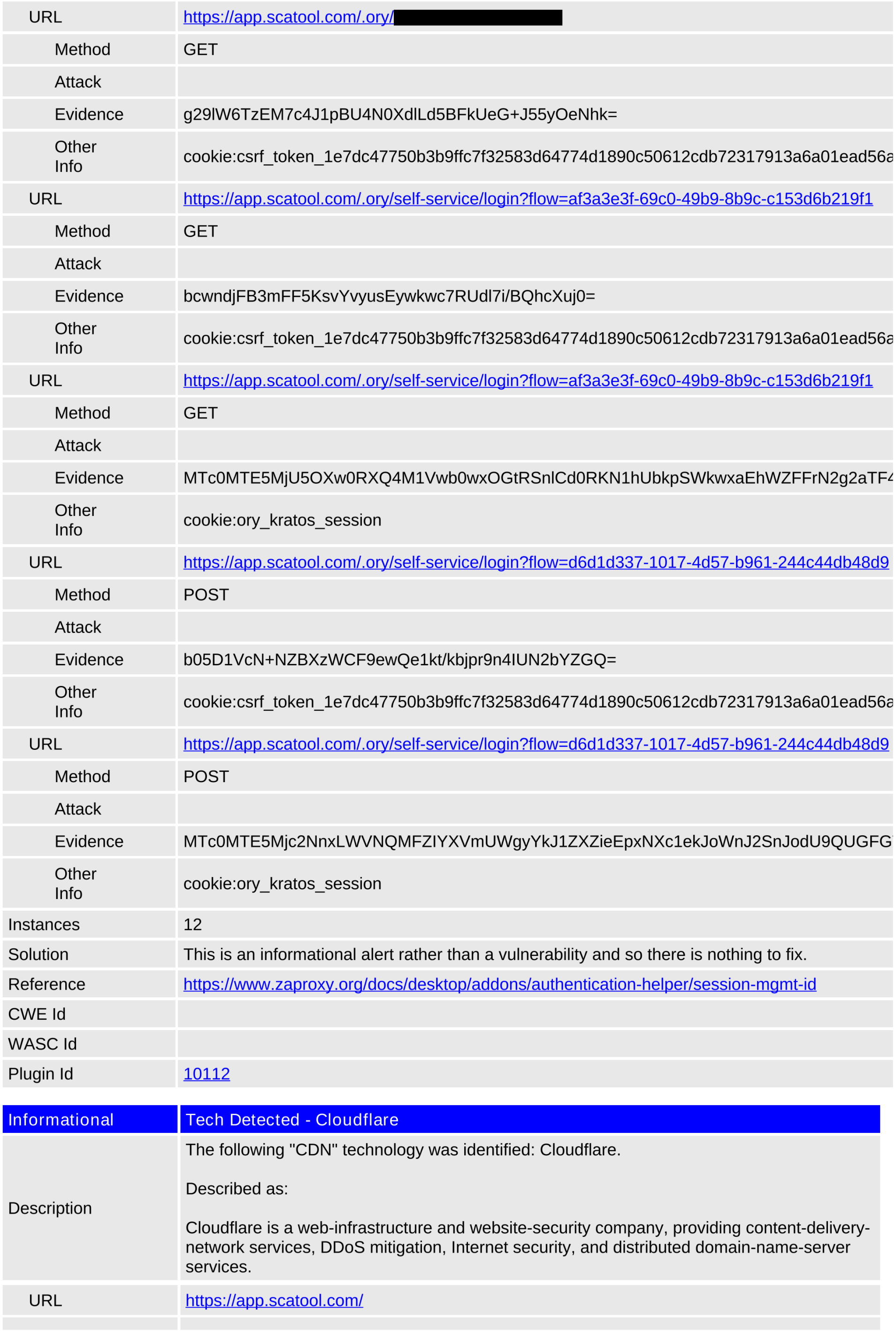
Informational	Re-examine Cache-control Directives
Description	The cache-control header has not been set properly or is missing, allowing the browser and proxies to cache content. For static assets like css, js, or image files this might be intended, however, the resources should be reviewed to ensure that no sensitive content will be cached.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://app.scatool.com/auth/login
Method	GET
Attack	
Evidence	
Other Info	
Instances	2
Solution	For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".
Reference	https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html#web-content-caching https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cache-Control https://grayduck.mn/2021/09/13/cache-control-recommendations/
CWE Id	525
WASC Id	13
Plugin Id	10015

Informational	Retrieved from Cache
Description	The content was retrieved from a shared cache. If the response data is sensitive, personal or user-specific, this may result in sensitive information being leaked. In some cases, this may even result in a user gaining complete control of the session of another user, depending on the configuration of the caching components in use in their environment. This is primarily an issue where caching servers such as "proxy" caches are configured on the local network. This configuration is typically found in corporate or educational environments, for instance.
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET



Attack	
Evidence	Age: 366
Other Info	The presence of the 'Age' header indicates that a HTTP/1.1 compliant caching server is in use.
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET
Attack	
Evidence	Age: 4333
Other Info	The presence of the 'Age' header indicates that a HTTP/1.1 compliant caching server is in use.
URL	https://app.scatool.com/assets/index-MFilYmfF.js
Method	GET
Attack	
Evidence	Age: 209
Other Info	The presence of the 'Age' header indicates that a HTTP/1.1 compliant caching server is in use.
URL	https://app.scatool.com/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	Age: 367
Other Info	The presence of the 'Age' header indicates that a HTTP/1.1 compliant caching server is in use.
URL	https://app.scatool.com/assets/inter-latin-wght-normal-C2S99t-D.woff2
Method	GET
Attack	
Evidence	Age: 4334
Other Info	The presence of the 'Age' header indicates that a HTTP/1.1 compliant caching server is in use.
Instances	5
Solution	Validate that the response does not contain sensitive, personal or user-specific information. If it does, consider the use of the following HTTP response headers, to limit, or prevent the content being stored and retrieved from the cache by another user: Cache-Control: no-cache, no-store, must-revalidate, private Pragma: no-cache Expires: 0 This configuration directs both HTTP 1.0 and HTTP 1.1 compliant caching servers to not store the response, and to not retrieve the response (without validation) from the cache, in response to a similar request.
Reference	https://tools.ietf.org/html/rfc7234 https://tools.ietf.org/html/rfc7231 https://www.rfc-editor.org/rfc/rfc9110.html
CWE Id	
WASC Id	
Plugin Id	10050

Informational	Session Management Response Identified
Description	The given response has been identified as containing a session management token. The 'Other
URL	https://app.scatool.com/.ory/[redacted]
Method	GET
Attack	
Evidence	g29lW6TzEM7c4J1pBU4N0XdlLd5BFkUeG+J55yOeNhk=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	/GdDFTBIQ4N10OkPTbud94kc4dOsHGZ+Ti/OlVT//ck=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	9nfXozsEy0oqNZa3wNPWXopOs1dva7hwUsVLzFR3JQw=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	CRDjSrr7xR5opZKDw+IiHv9g1IHAJ0seLV0ij6UU2/g=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	
Evidence	JQAzxNsP+MRKDCBlsxJ8XHsZCqiWfCk1za9RAZ/FvD4=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	POST
Attack	
Evidence	bcwndjFB3mFF5KsvYvyusEywkwc7RUdl7i/BQhcXuj0=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	b05D1VcN+NZBXzWCF9ewQe1kt/kbjpr9n4IUN2bYZGQ=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a



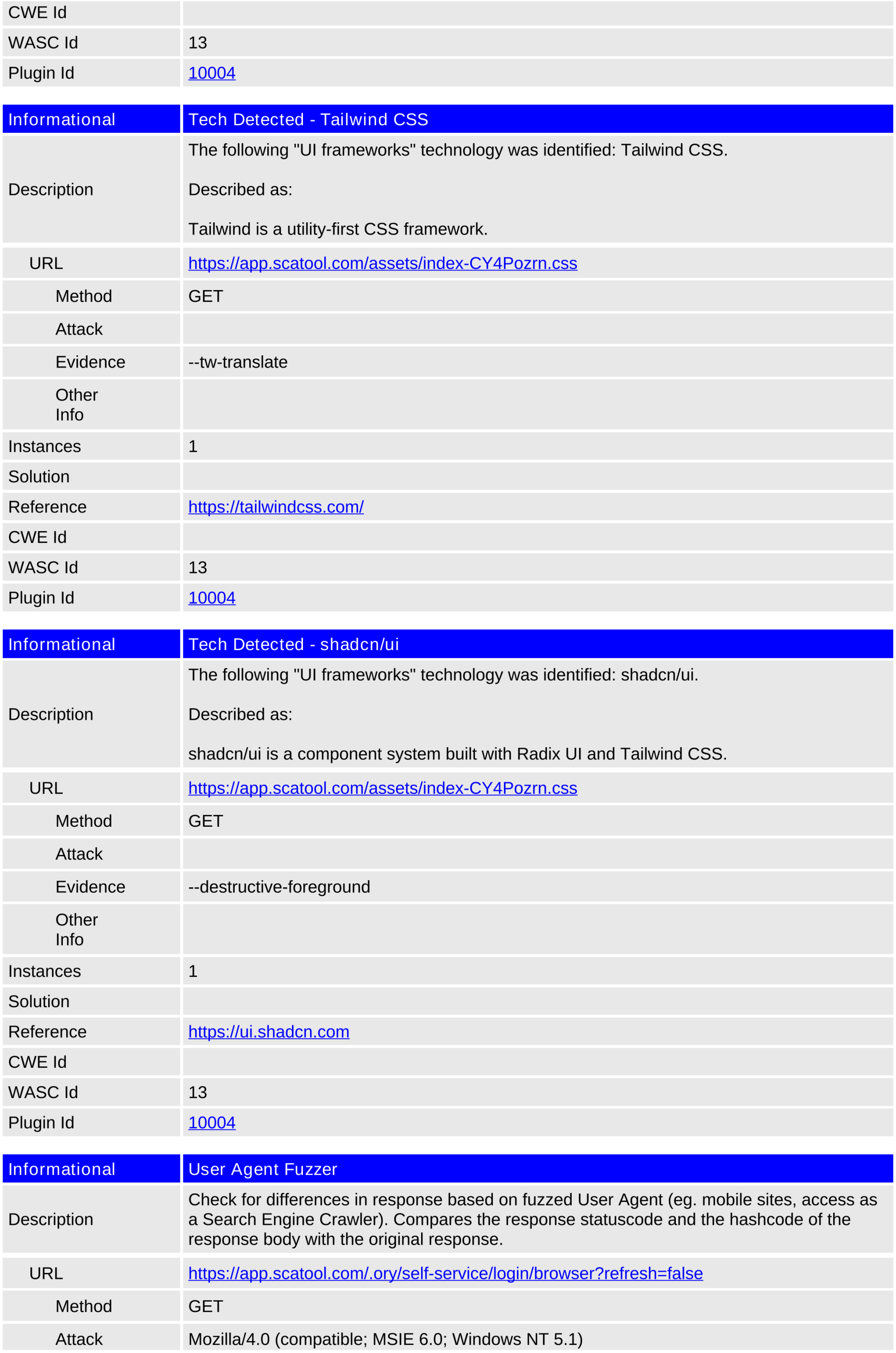
URL	https://app.scatool.com/.ory/[redacted]
Method	GET
Attack	
Evidence	g29lW6TzEM7c4J1pBU4N0XdlLd5BFkUeG+J55yOeNhk=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	GET
Attack	
Evidence	bcwndjFB3mFF5KsvYvyusEywkwc7RUdl7i/BQhcXuj0=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login?flow=af3a3e3f-69c0-49b9-8b9c-c153d6b219f1
Method	GET
Attack	
Evidence	MTc0MTE5MjU5OXw0RXQ4M1Vwb0wxOGtRSnlCd0RKN1hUbkpSWkwxaEhWZFFrN2g2aTF4
Other Info	cookie:ory_kratos_session
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	b05D1VcN+NZBXzWCF9ewQe1kt/kbjpr9n4IUN2bYZGQ=
Other Info	cookie:csrf_token_1e7dc47750b3b9ffc7f32583d64774d1890c50612cdb72317913a6a01ead56a
URL	https://app.scatool.com/.ory/self-service/login?flow=d6d1d337-1017-4d57-b961-244c44db48d9
Method	POST
Attack	
Evidence	MTc0MTE5Mjc2NnxLWVNQMFZIYXVmUWgyYkJ1ZXZieEpxNXc1ekJoWnJ2SnJodU9QUGFG
Other Info	cookie:ory_kratos_session
Instances	12
Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.
Reference	https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id
CWE Id	
WASC Id	
Plugin Id	10112

Informational	Tech Detected - Cloudflare
Description	The following "CDN" technology was identified: Cloudflare. Described as: Cloudflare is a web-infrastructure and website-security company, providing content-delivery-network services, DDoS mitigation, Internet security, and distributed domain-name-server services.
URL	https://app.scatool.com/

Method	GET
Attack	
Evidence	cf-ray
Other Info	The following CPE is associated with the identified tech: cpe:2.3:a:cloudflare:cloudflare:*:*:*:*:*:*:*:*
Instances	1
Solution	
Reference	https://www.cloudflare.com
CWE Id	
WASC Id	13
Plugin Id	10004

Informational	Tech Detected - HTTP/3
Description	The following "Miscellaneous" technology was identified: HTTP/3. Described as: HTTP/3 is the third major version of the Hypertext Transfer Protocol used to exchange information on the World Wide Web.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	h3
Other Info	
Instances	1
Solution	
Reference	https://httpwg.org/
CWE Id	
WASC Id	13
Plugin Id	10004

Informational	Tech Detected - PWA
Description	The following "Miscellaneous" technology was identified: PWA. Described as: Progressive Web Apps (PWAs) are web apps built and enhanced with modern APIs to deliver enhanced capabilities, reliability, and installability while reaching anyone, anywhere, on any device, all with a single codebase.
URL	https://app.scatool.com/
Method	GET
Attack	
Evidence	
Other Info	
Instances	1
Solution	
Reference	https://web.dev/progressive-web-apps/



CWE Id	
WASC Id	13
Plugin Id	10004

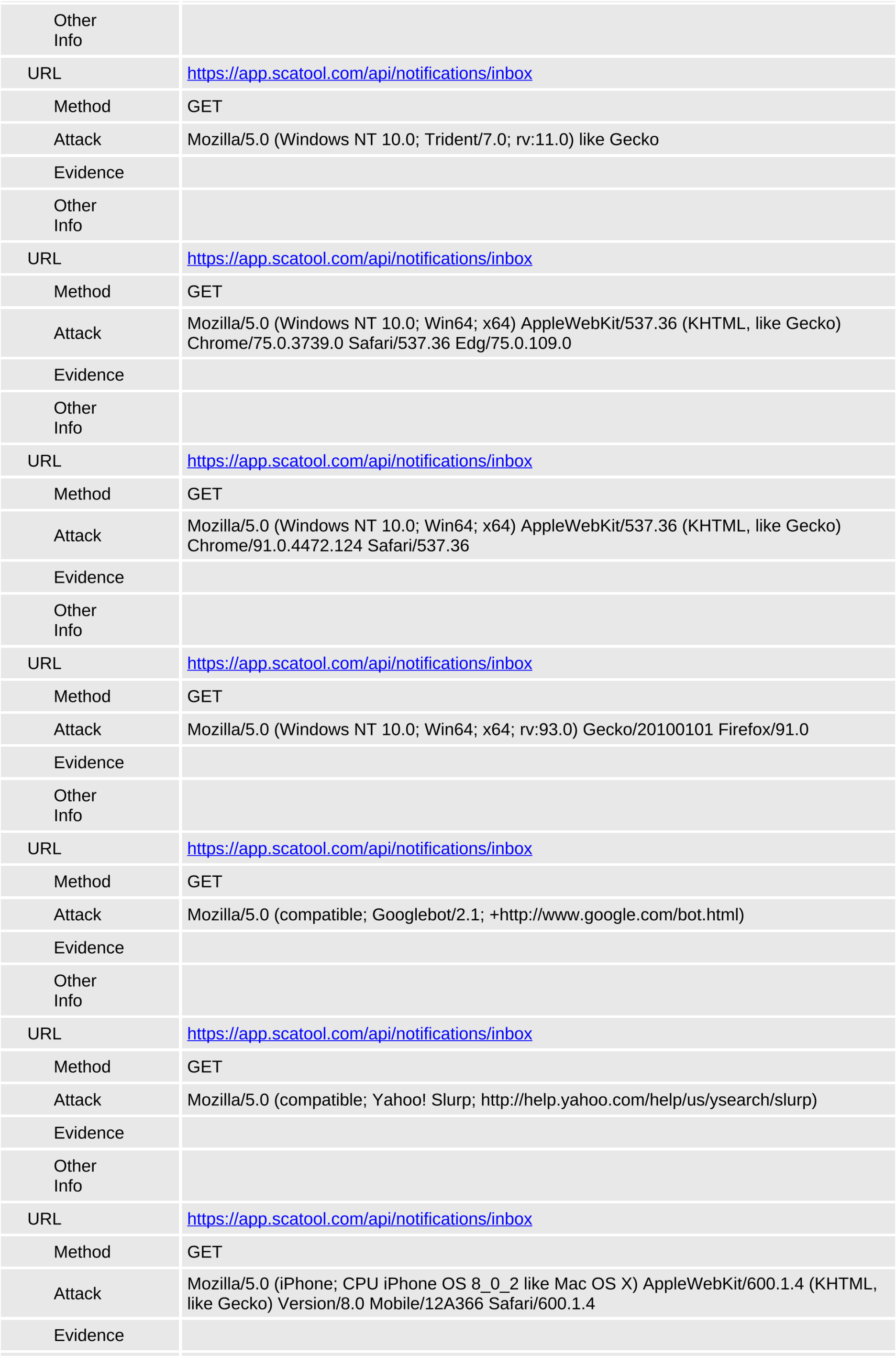
Informational	Tech Detected - Tailwind CSS
Description	The following "UI frameworks" technology was identified: Tailwind CSS. Described as: Tailwind is a utility-first CSS framework.
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET
Attack	
Evidence	--tw-translate
Other Info	
Instances	1
Solution	
Reference	https://tailwindcss.com/
CWE Id	
WASC Id	13
Plugin Id	10004

Informational	Tech Detected - shadcn/ui
Description	The following "UI frameworks" technology was identified: shadcn/ui. Described as: shadcn/ui is a component system built with Radix UI and Tailwind CSS.
URL	https://app.scatool.com/assets/index-CY4Pozrn.css
Method	GET
Attack	
Evidence	--destructive-foreground
Other Info	
Instances	1
Solution	
Reference	https://ui.shadcn.com
CWE Id	
WASC Id	13
Plugin Id	10004

Informational	User Agent Fuzzer
Description	Check for differences in response based on fuzzed User Agent (eg. mobile sites, access as a Search Engine Crawler). Compares the response statuscode and the hashcode of the response body with the original response.
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)

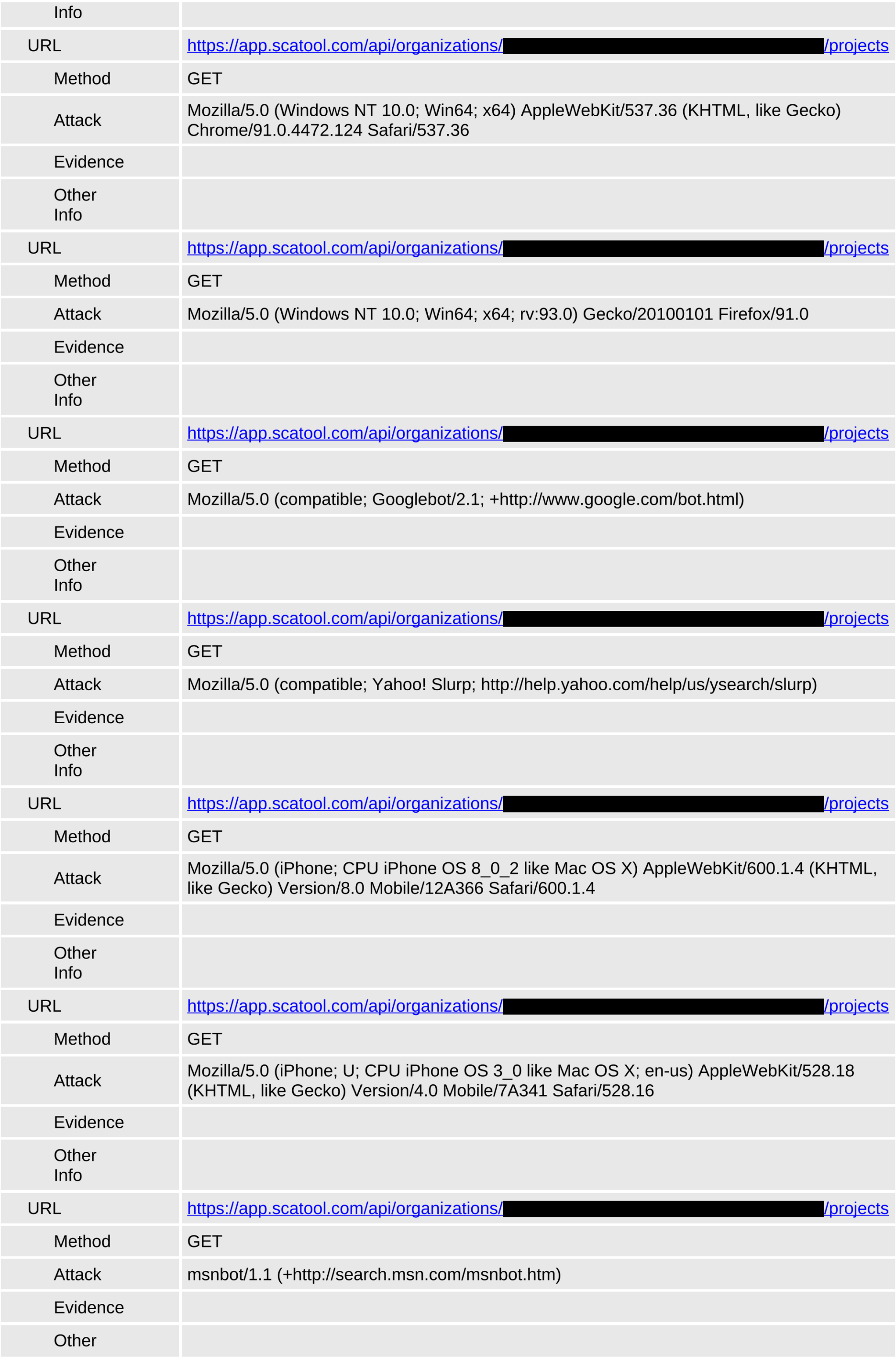
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)

Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	https://app.scatool.com/.ory/self-service/login/browser?refresh=false
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	



Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	

Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	https://app.scatool.com/api/notifications/inbox
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other	

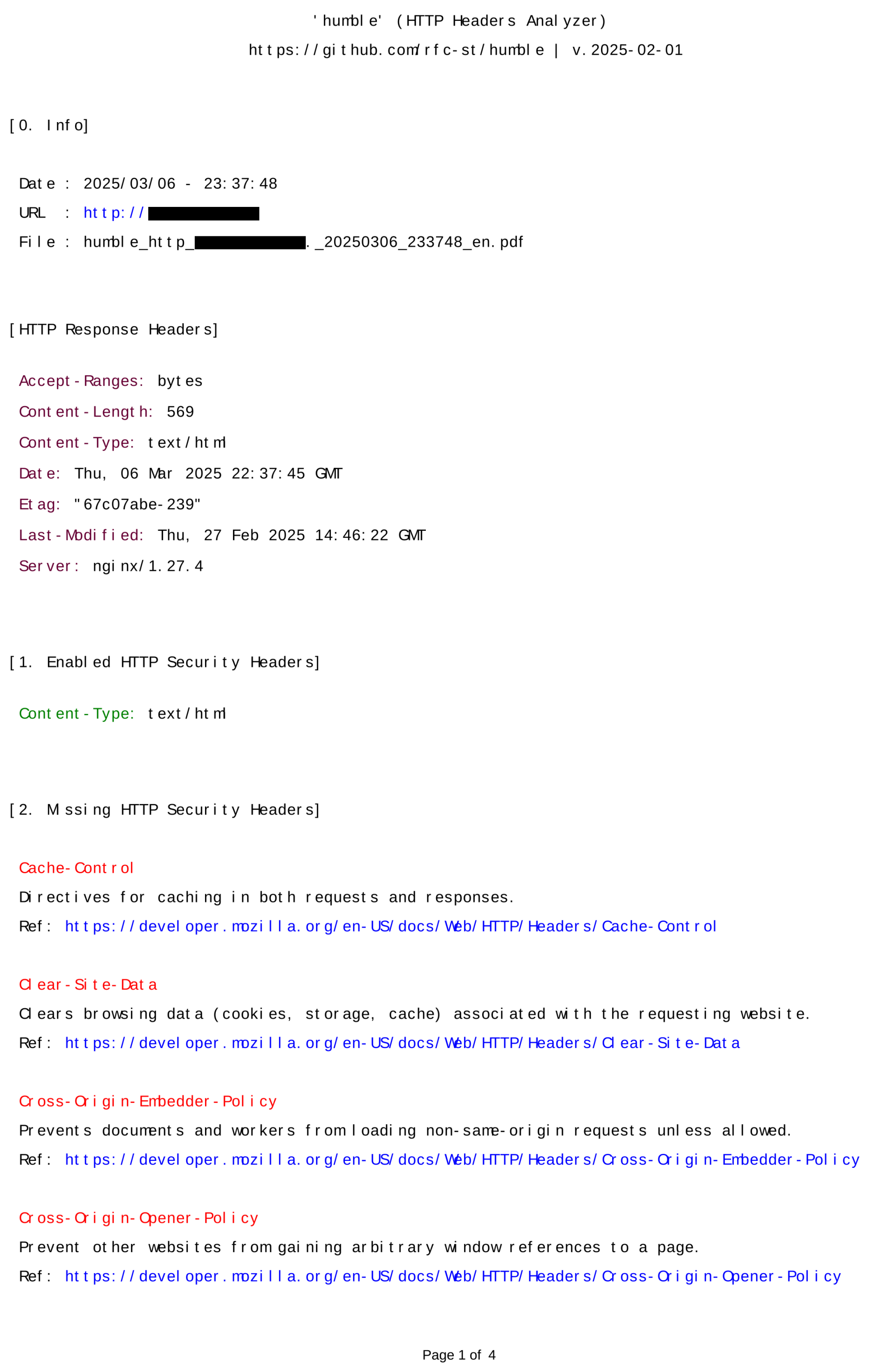


Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	https://app.scatool.com/api/organizations/[redacted]/projects
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other	

Info	
Instances	36
Solution	
Reference	https://owasp.org/wstg
CWE Id	
WASC Id	
Plugin Id	10104

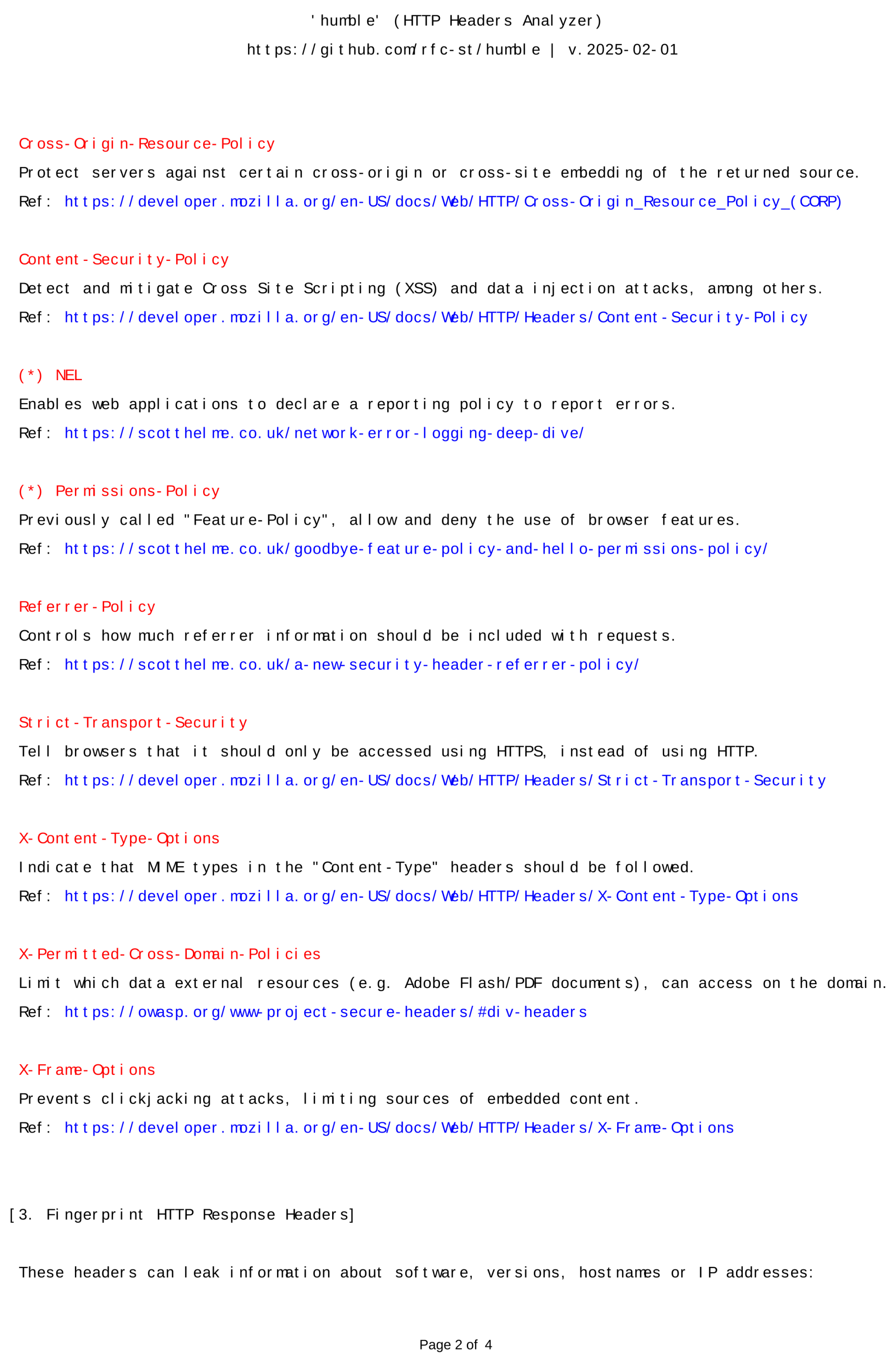
C *humble* Report of *SCA Tool Application* - REDACTED (testing)

The *humble* report of the test run against `testing` can be found on the following pages.



```
'humble' (HTTP Headers Analyzer)
https://github.com/rfc-st/humble | v.2025-02-01

[0. Info]

 Date : 2025/03/06 - 23:37:48
 URL  : http://[redacted]
 File : humble_http_[redacted]._20250306_233748_en.pdf


[HTTP Response Headers]

 Accept-Ranges: bytes
 Content-Length: 569
 Content-Type: text/html
 Date: Thu, 06 Mar 2025 22:37:45 GMT
 Etag: "67c07abe-239"
 Last-Modified: Thu, 27 Feb 2025 14:46:22 GMT
 Server: nginx/1.27.4


[1. Enabled HTTP Security Headers]

 Content-Type: text/html


[2. Missing HTTP Security Headers]

 Cache-Control
 Directives for caching in both requests and responses.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cache-Control

 Clear-Site-Data
 Clears browsing data (cookies, storage, cache) associated with the requesting website.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Clear-Site-Data

 Cross-Origin-Embedder-Policy
 Prevents documents and workers from loading non-same-origin requests unless allowed.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Embedder-Policy

 Cross-Origin-Opener-Policy
 Prevent other websites from gaining arbitrary window references to a page.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Opener-Policy
```

Page 1 of 4



'humble' (HTTP Headers Analyzer)
https://github.com/rfc-st/humble | v.2025-02-01

Cross-Origin-Resource-Policy
Protect servers against certain cross-origin or cross-site embedding of the returned source.
Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cross-Origin_Resource_Policy_(CORP)

Content-Security-Policy
Detect and mitigate Cross Site Scripting (XSS) and data injection attacks, among others.
Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy

(*) NEL
Enables web applications to declare a reporting policy to report errors.
Ref: https://scotthelme.co.uk/network-error-logging-deep-dive/

(*) Permissions-Policy
Previously called "Feature-Policy", allow and deny the use of browser features.
Ref: https://scotthelme.co.uk/goodbye-feature-policy-and-hello-permissions-policy/

Referrer-Policy
Controls how much referrer information should be included with requests.
Ref: https://scotthelme.co.uk/a-new-security-header-referrer-policy/

Strict-Transport-Security
Tell browsers that it should only be accessed using HTTPS, instead of using HTTP.
Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security

X-Content-Type-Options
Indicate that MIME types in the "Content-Type" headers should be followed.
Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

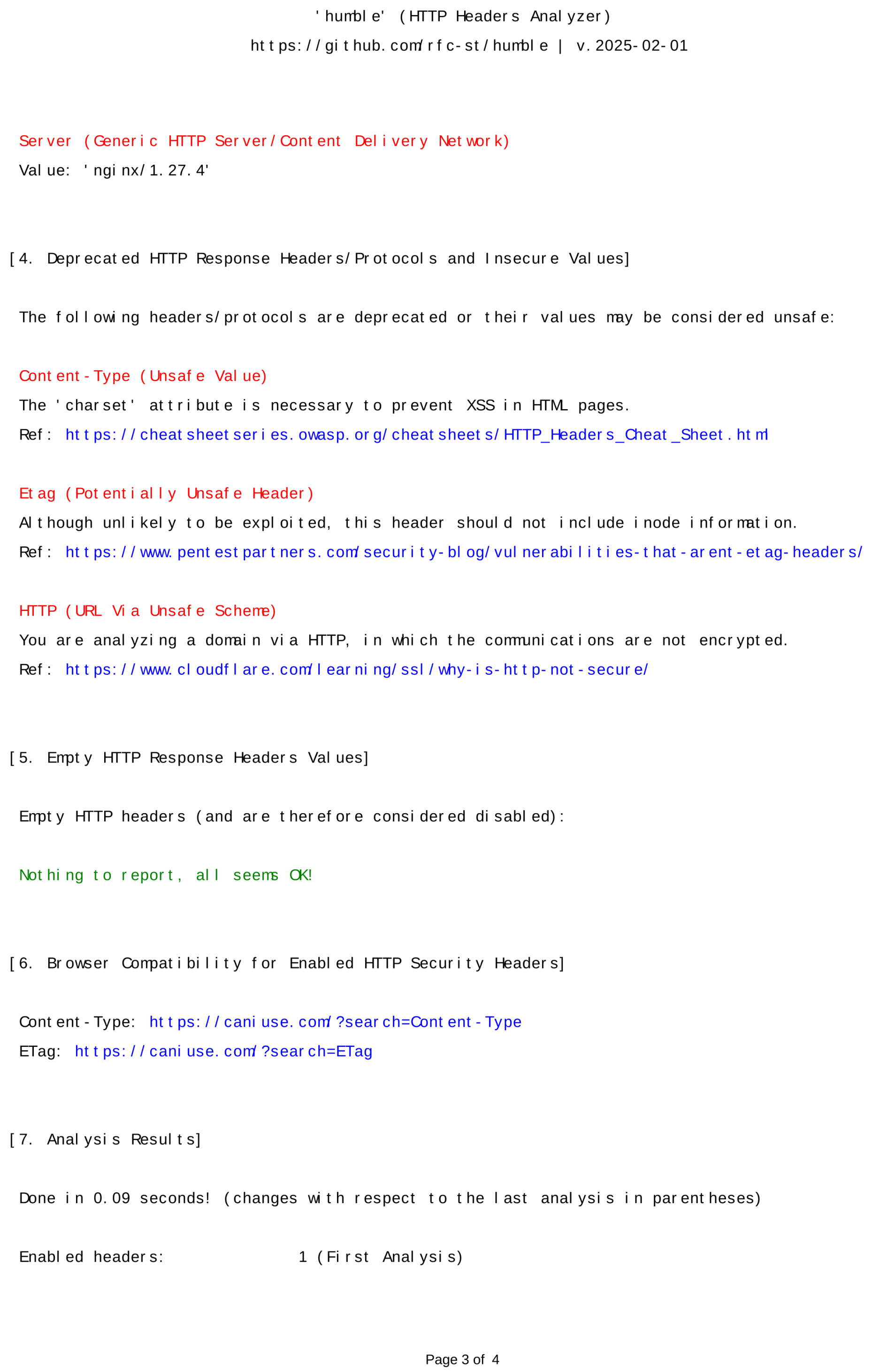
X-Permitted-Cross-Domain-Policies
Limit which data external resources (e.g. Adobe Flash/PDF documents), can access on the domain.
Ref: https://owasp.org/www-project-secure-headers/#div-headers

X-Frame-Options
Prevents clickjacking attacks, limiting sources of embedded content.
Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options

[3. Fingerprint HTTP Response Headers]

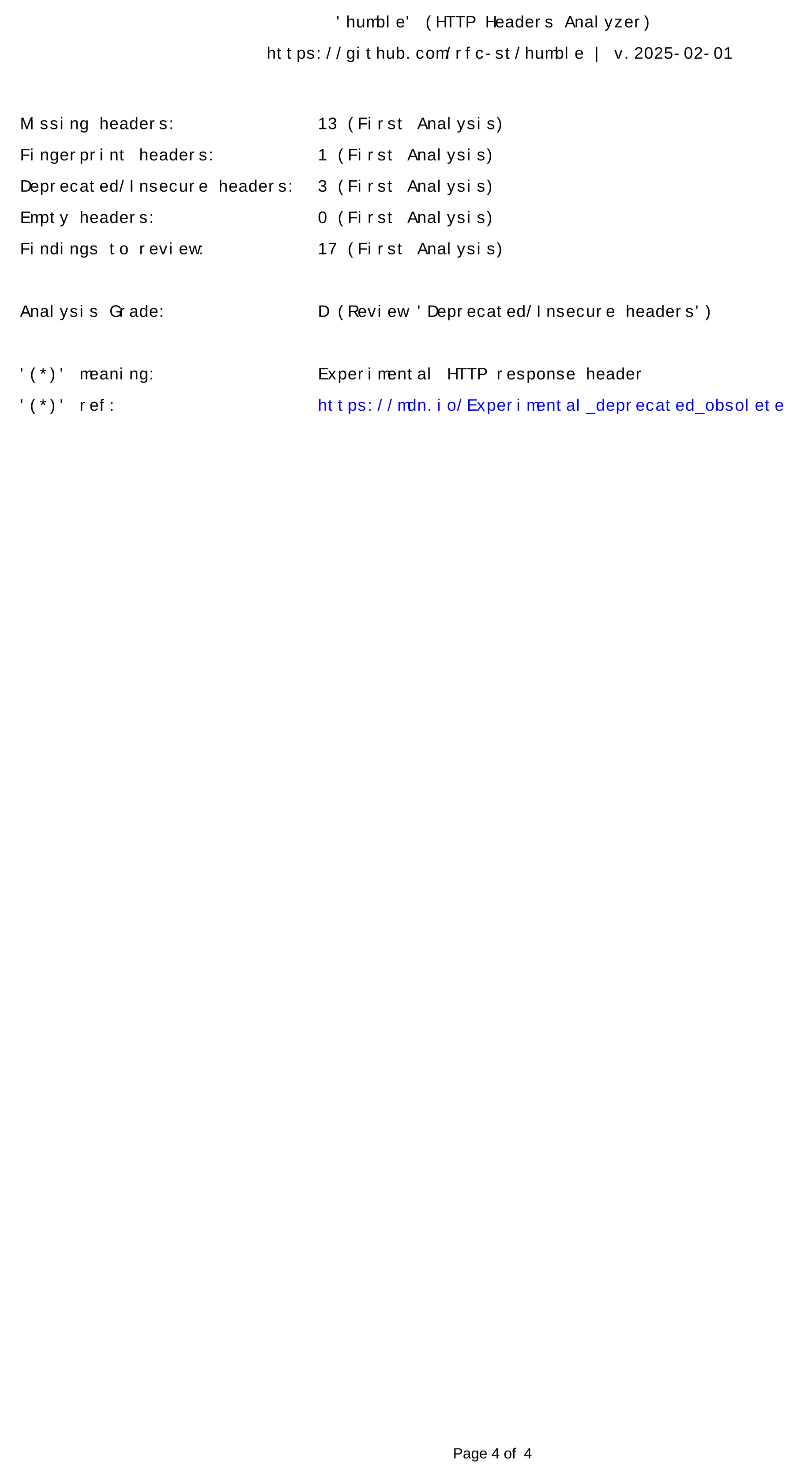
These headers can leak information about software, versions, hostnames or IP addresses:

Page 2 of 4



```
                    'humble' (HTTP Headers Analyzer)
               https://github.com/rfc-st/humble | v.2025-02-01


 Server (Generic HTTP Server/Content Delivery Network)
 Value: 'nginx/1.27.4'


[4. Deprecated HTTP Response Headers/Protocols and Insecure Values]

 The following headers/protocols are deprecated or their values may be considered unsafe:

 Content-Type (Unsafe Value)
 The 'charset' attribute is necessary to prevent XSS in HTML pages.
 Ref: https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Headers_Cheat_Sheet.html

 Etag (Potentially Unsafe Header)
 Although unlikely to be exploited, this header should not include inode information.
 Ref: https://www.pentestpartners.com/security-blog/vulnerabilities-that-arent-etag-headers/

 HTTP (URL Via Unsafe Scheme)
 You are analyzing a domain via HTTP, in which the communications are not encrypted.
 Ref: https://www.cloudflare.com/learning/ssl/why-is-http-not-secure/


[5. Empty HTTP Response Headers Values]

 Empty HTTP headers (and are therefore considered disabled):

 Nothing to report, all seems OK!


[6. Browser Compatibility for Enabled HTTP Security Headers]

 Content-Type: https://caniuse.com/?search=Content-Type
 ETag: https://caniuse.com/?search=ETag


[7. Analysis Results]

 Done in 0.09 seconds! (changes with respect to the last analysis in parentheses)

 Enabled headers:              1 (First Analysis)


                              Page 3 of 4
```

Appendix C: *humble* Report of *SCA Tool Application* - REDACTED (testing)



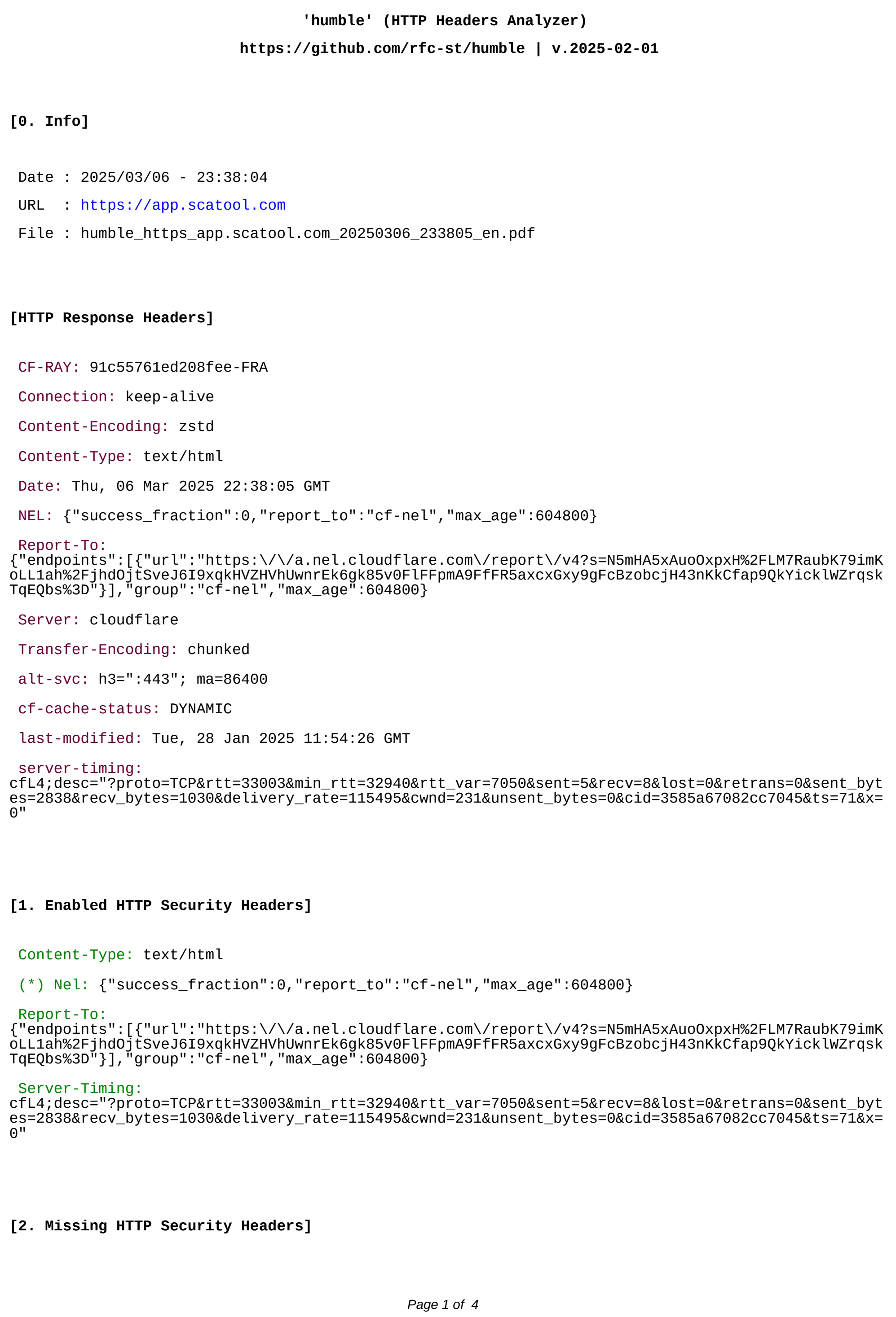
```
                    'humble' (HTTP Headers Analyzer)
           https://github.com/rfc-st/humble | v.2025-02-01

 Missing headers:               13 (First Analysis)
 Fingerprint headers:           1 (First Analysis)
 Deprecated/Insecure headers:   3 (First Analysis)
 Empty headers:                 0 (First Analysis)
 Findings to review:            17 (First Analysis)

 Analysis Grade:                D (Review 'Deprecated/Insecure headers')

 '(*)' meaning:                 Experimental HTTP response header
 '(*)' ref:                     https://mdn.io/Experimental_deprecated_obsolete
```

Page 4 of 4

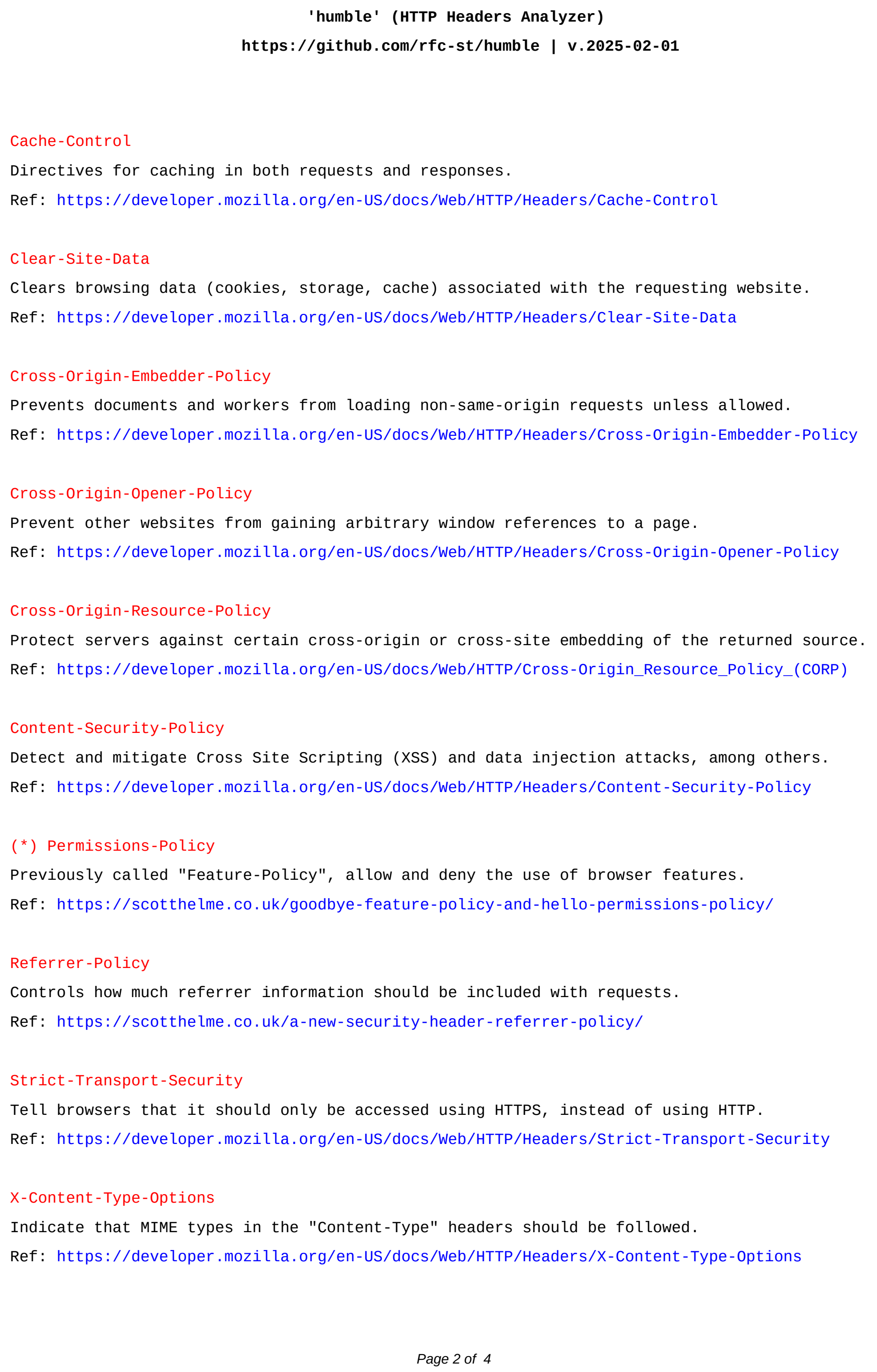
D *humble* Report of *SCA Tool Application* - app.scatool.com (production)

The *humble* report of the test run against `production` can be found on the following pages.

Appendix D: *humble* Report of *SCA Tool Application* - app.scatool.com (production)



```
'humble' (HTTP Headers Analyzer)
https://github.com/rfc-st/humble | v.2025-02-01


[0. Info]

 Date : 2025/03/06 - 23:38:04
 URL  : https://app.scatool.com
 File : humble_https_app.scatool.com_20250306_233805_en.pdf


[HTTP Response Headers]

 CF-RAY: 91c55761ed208fee-FRA
 Connection: keep-alive
 Content-Encoding: zstd
 Content-Type: text/html
 Date: Thu, 06 Mar 2025 22:38:05 GMT
 NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
 Report-To:
{"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v4?s=N5mHA5xAuoOxpxH%2FLM7RaubK79imK
oLL1ah%2FjhdOjtSveJ6I9xqkHVZHVhUwnrEk6gk85v0FlFFpmA9FfFR5axcxGxy9gFcBzobcjH43nKkCfap9QkYicklWZrqsk
TqEQbs%3D"}],"group":"cf-nel","max_age":604800}
 Server: cloudflare
 Transfer-Encoding: chunked
 alt-svc: h3=":443"; ma=86400
 cf-cache-status: DYNAMIC
 last-modified: Tue, 28 Jan 2025 11:54:26 GMT
 server-timing:
cfL4;desc="?proto=TCP&rtt=33003&min_rtt=32940&rtt_var=7050&sent=5&recv=8&lost=0&retrans=0&sent_byt
es=2838&recv_bytes=1030&delivery_rate=115495&cwnd=231&unsent_bytes=0&cid=3585a67082cc7045&ts=71&x=
0"


[1. Enabled HTTP Security Headers]

 Content-Type: text/html
 (*) Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
 Report-To:
{"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v4?s=N5mHA5xAuoOxpxH%2FLM7RaubK79imK
oLL1ah%2FjhdOjtSveJ6I9xqkHVZHVhUwnrEk6gk85v0FlFFpmA9FfFR5axcxGxy9gFcBzobcjH43nKkCfap9QkYicklWZrqsk
TqEQbs%3D"}],"group":"cf-nel","max_age":604800}
 Server-Timing:
cfL4;desc="?proto=TCP&rtt=33003&min_rtt=32940&rtt_var=7050&sent=5&recv=8&lost=0&retrans=0&sent_byt
es=2838&recv_bytes=1030&delivery_rate=115495&cwnd=231&unsent_bytes=0&cid=3585a67082cc7045&ts=71&x=
0"


[2. Missing HTTP Security Headers]
```



'humble' (HTTP Headers Analyzer)

https://github.com/rfc-st/humble | v.2025-02-01

Cache-Control

Directives for caching in both requests and responses.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cache-Control

Clear-Site-Data

Clears browsing data (cookies, storage, cache) associated with the requesting website.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Clear-Site-Data

Cross-Origin-Embedder-Policy

Prevents documents and workers from loading non-same-origin requests unless allowed.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Embedder-Policy

Cross-Origin-Opener-Policy

Prevent other websites from gaining arbitrary window references to a page.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Opener-Policy

Cross-Origin-Resource-Policy

Protect servers against certain cross-origin or cross-site embedding of the returned source.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cross-Origin_Resource_Policy_(CORP)

Content-Security-Policy

Detect and mitigate Cross Site Scripting (XSS) and data injection attacks, among others.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy

(*) Permissions-Policy

Previously called "Feature-Policy", allow and deny the use of browser features.

Ref: https://scotthelme.co.uk/goodbye-feature-policy-and-hello-permissions-policy/

Referrer-Policy

Controls how much referrer information should be included with requests.

Ref: https://scotthelme.co.uk/a-new-security-header-referrer-policy/

Strict-Transport-Security

Tell browsers that it should only be accessed using HTTPS, instead of using HTTP.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security

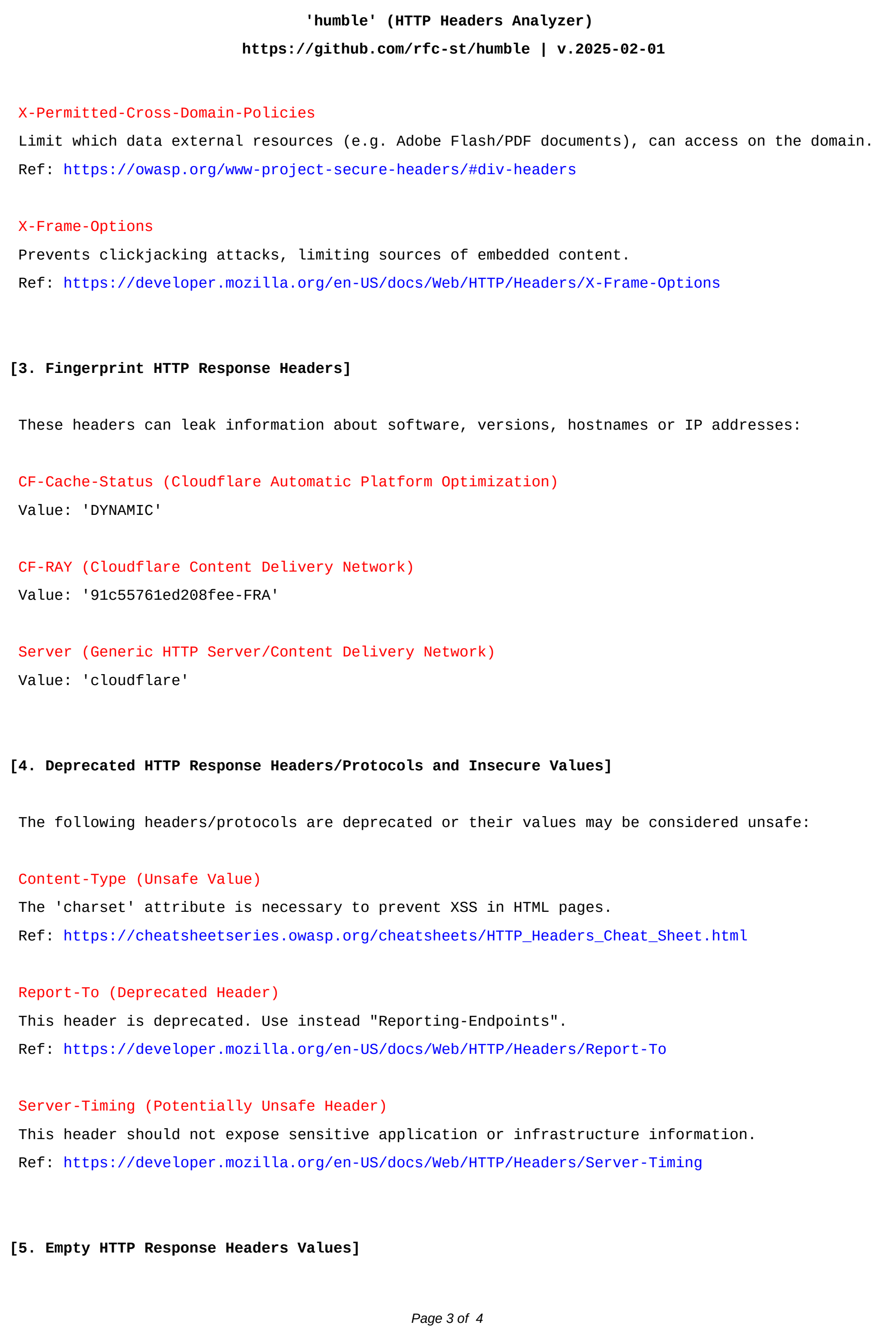
X-Content-Type-Options

Indicate that MIME types in the "Content-Type" headers should be followed.

Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

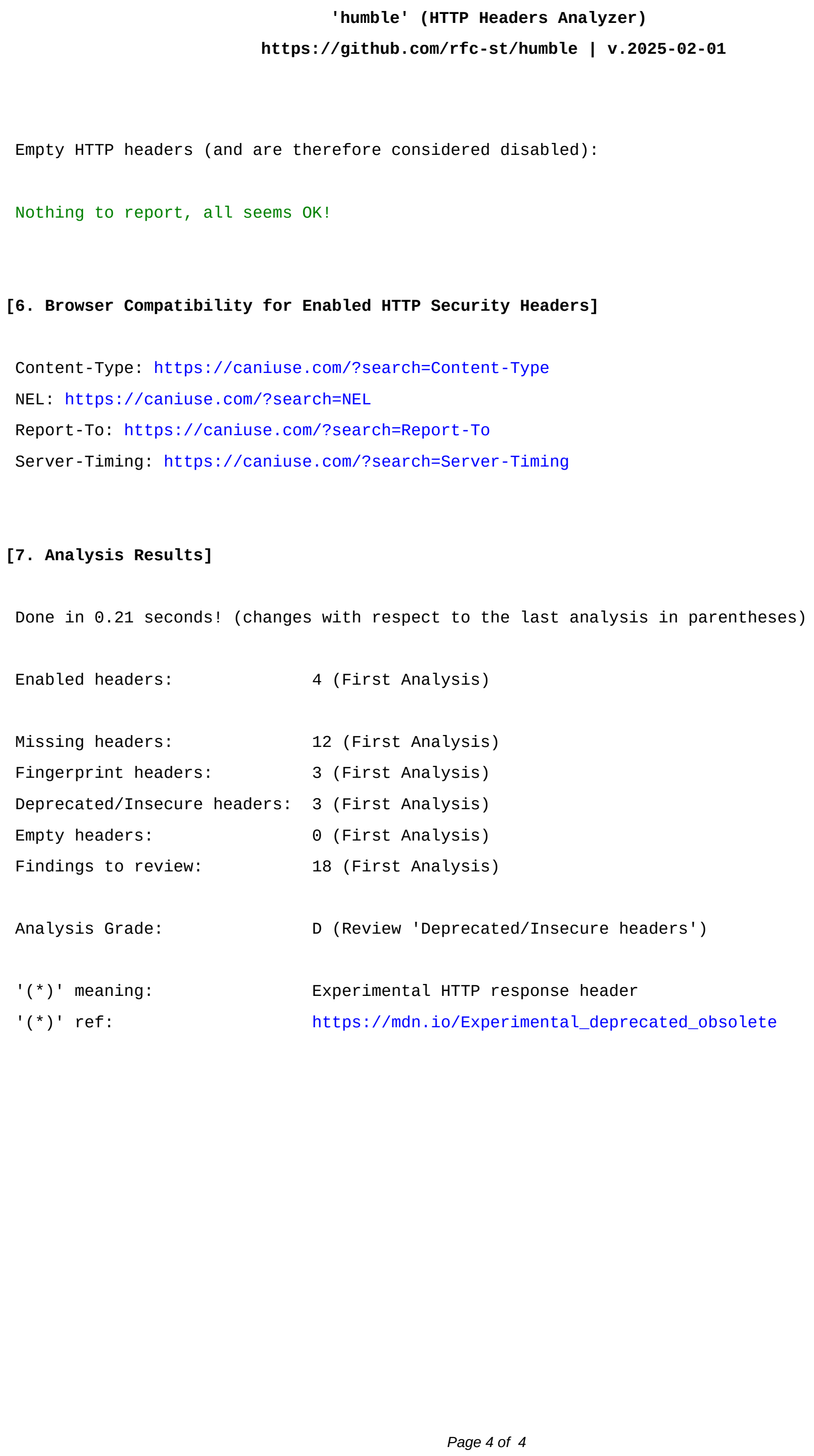
Page 2 of 4

Appendix D: *humble* Report of *SCA Tool Application* - app.scatool.com (production)



```
                  'humble' (HTTP Headers Analyzer)
          https://github.com/rfc-st/humble | v.2025-02-01

 X-Permitted-Cross-Domain-Policies
 Limit which data external resources (e.g. Adobe Flash/PDF documents), can access on the domain.
 Ref: https://owasp.org/www-project-secure-headers/#div-headers

 X-Frame-Options
 Prevents clickjacking attacks, limiting sources of embedded content.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options


[3. Fingerprint HTTP Response Headers]

 These headers can leak information about software, versions, hostnames or IP addresses:

 CF-Cache-Status (Cloudflare Automatic Platform Optimization)
 Value: 'DYNAMIC'

 CF-RAY (Cloudflare Content Delivery Network)
 Value: '91c55761ed208fee-FRA'

 Server (Generic HTTP Server/Content Delivery Network)
 Value: 'cloudflare'


[4. Deprecated HTTP Response Headers/Protocols and Insecure Values]

 The following headers/protocols are deprecated or their values may be considered unsafe:

 Content-Type (Unsafe Value)
 The 'charset' attribute is necessary to prevent XSS in HTML pages.
 Ref: https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Headers_Cheat_Sheet.html

 Report-To (Deprecated Header)
 This header is deprecated. Use instead "Reporting-Endpoints".
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Report-To

 Server-Timing (Potentially Unsafe Header)
 This header should not expose sensitive application or infrastructure information.
 Ref: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Server-Timing


[5. Empty HTTP Response Headers Values]
```

Page 3 of 4



```
                    'humble' (HTTP Headers Analyzer)
             https://github.com/rfc-st/humble | v.2025-02-01


 Empty HTTP headers (and are therefore considered disabled):

 Nothing to report, all seems OK!


[6. Browser Compatibility for Enabled HTTP Security Headers]

 Content-Type: https://caniuse.com/?search=Content-Type
 NEL: https://caniuse.com/?search=NEL
 Report-To: https://caniuse.com/?search=Report-To
 Server-Timing: https://caniuse.com/?search=Server-Timing


[7. Analysis Results]

 Done in 0.21 seconds! (changes with respect to the last analysis in parentheses)

 Enabled headers:              4 (First Analysis)

 Missing headers:              12 (First Analysis)
 Fingerprint headers:          3 (First Analysis)
 Deprecated/Insecure headers:  3 (First Analysis)
 Empty headers:                0 (First Analysis)
 Findings to review:           18 (First Analysis)

 Analysis Grade:               D (Review 'Deprecated/Insecure headers')

 '(*)' meaning:                Experimental HTTP response header
 '(*)' ref:                    https://mdn.io/Experimental_deprecated_obsolete
```

Page 4 of 4

E Master's Thesis - Digital Version

This thesis is available as a PDF file on the attached Compact Disc (CD) below.

References

Al Fardan, N. J., & Paterson, K. G. (2013). Lucky thirteen: Breaking the TLS and DTLS record protocols. *2013 IEEE Symposium on Security and Privacy*, 526–540. https://doi.org/10.1109/SP.2013.42

Aumasson, J.-P. (2019). *Password hashing competition.* https://www.password-hashing.net/

Biryukov, A., Dinu, D., & Khovratovich, D. (2017). argon2: The memory-hard function for password hashing and other applications. https://github.com/P-H-C/phc-winner-argon2/blob/f57e61e19229e23c4445b85494dbf7c07de721cb/argon2-specs.pdf

Brenner, M., Gentschen Felde, N., Hommel, W., Metzger, S., Reiser, H., & Schaaf, T. (2024, July 3). Praxisbuch ISO/IEC 27001. In *Praxisbuch ISO/IEC 27001* (pp. I–XIV). Carl Hanser Verlag GmbH & Co. KG. https://doi.org/10.3139/9783446478459.fm

Bruma, L. M. (2021). Cloud security audit – issues and challenges. *2021 16th International Conference on Computer Science & Education (ICCSE).* https://doi.org/10.1109/iccse51940.2021.9569654

BSI. (2017a, October). BSI-Standard 200-1 - Managementsysteme für Informationssicherheit (ISMS). https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/BSI_Standards/standard_200_1.pdf

BSI. (2017b, October). BSI-standard 200-3 - risk analysis based on IT-grundschutz. https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Grundschutz/International/bsi-standard-2003_en_pdf.pdf?__blob=publicationFile&v=2

Casola, V., De Benedictis, A., Mazzocca, C., & Orbinato, V. (2024). Secure software development and testing: A model-based methodology. *Computers & Security*, *137*, 103639. https://doi.org/10.1016/j.cose.2023.103639

Cloudflare, Inc. (2021, April 20). How the cloudflare network maintains data privacy. https://www.cloudflare.com/resources/assets/slt3lc6tev37/1S0GCmfDE4rA1CK5dCK9kL/1410444a21bedc1aefaa9e332b4a1e34/Data_Transit_Privacy_Whitepaper_V2.pdf

Discord, Inc. (2023, March 15). *How long discord keeps your information* [How long discord keeps your information]. Retrieved February 1, 2025, from https://support.discord.com/hc/en-us/articles/5431812448791-How-long-Discord-keeps-your-information

Discord, Inc. (2024, April 15). *Privacy policy | discord* [Discord privacy policy]. Retrieved February 1, 2025, from https://discord.com/privacy#5

Discord, Inc. (2025, March 17). *About discord | our mission and story* [Discord]. Retrieved March 17, 2025, from https://discord.com/company

Dr. Johann Schlamp, Prof. Dr. Thomas C. Schmidt & Prof. Dr. Matthias Wählisch. (2022, February 21). *Zweite Internet Backbone-Studie: Auslandskabelverbindungen und CDN-Kompetenz (ZwIBACK)* (Projekt 415 Los 1). Leitwert GmbH. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/ZwiBACK/ZwiBACK-Studie.pdf

Düsterhus, T. (2023, September 7). *PHP RFC: Increasing the default BCrypt cost* [PHP RFC: Increasing the default BCrypt cost]. Retrieved March 15, 2025, from https://wiki.php.net/rfc/bcrypt_cost_2023

European Parliament and Council of European Union. (2024). Regulation (EU) 2024/2847 of the european parliament and of the council of 23 october 2024 on horizontal cybersecurity requirements for products with digital elements and amending regulations (EU) no 168/2013 and (EU) no 2019/1020 and directive (EU) 2020/1828 (cyber resilience act) (text with EEA relevance). *Document 32024R2847*. Retrieved March 9, 2025, from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32024R2847

European Parliament and Council of the European Union. (2016, April 27). REGULATION (EU) 2016/ 679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL - of 27 april 2016 - on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/ 46/ EC (general data protection regulation). https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679

Fraunhofer SIT. (2018, August 7). IT-grundschutz online course. Retrieved February 9, 2025, from https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Grundschutz/Webkurs/IT_Grundschutz_Online_Course.pdf

Friedrlch-Alexander-University Erlangen-Nuremberg. (2023, December 1). *FAU - facts and figures* [Facts and figures]. Retrieved March 12, 2025, from https://www.fau.eu/fau/welcome-to-fau/facts-and-figures/

GitHub, Inc. (2025, March 9). *GitHub terms of service* [GitHub docs]. Retrieved March 9, 2025, from https://docs-internal.github.com/en/site-policy/github-terms/github-terms-of-service

Jadhav, K. (2023, January 31). *THE ROLE OF CYBER SECURITY AUDITS*.

Jover, R. P. (2020). Security analysis of SMS as a second factor of authentication: The challenges of multifactor authentication based on SMS, including cellular security deficiencies, SS7 exploits, and SIM swapping. *Queue*, *18*(4), 37–60. https://doi.org/10.1145/3424302.3425909

Karim, N. A., Kanaker, H., Abdulraheem, W. K., Ghaith, M. A., Alhroob, E., & Alali, A. M. F. (2024). Choosing the right MFA method for online systems: A comparative analysis. *International Journal of Data and Network Science*, *8*(1), 201–212. https://doi.org/10.5267/j.ijdns.2023.10.003

Kuketz, M. (2020, February 4). *Threema: Instant-messaging-dienst aus der schweiz – messenger teil2* [Kuketz-blog]. Retrieved March 15, 2025, from https://www.kuketz-blog.de/threema-instant-messaging-dienst-aus-der-schweiz-messenger-teil2/

Kuketz, M. (2024, January 11). *Jenseits der grenzen: Überblick über das US-geheimdienstrecht* [Kuketz-blog]. Retrieved March 17, 2025, from https://www.kuketz-blog.de/jenseits-der-grenzen-ueberblick-ueber-das-us-geheimdienstrecht/

Moriarty, K., & Farrell, S. (2021, March). Deprecating TLS 1.0 and TLS 1.1. https://doi.org/10.17487/RFC8996

Nas, S., & Terra, F. (2021, February 12). DPIA google g suite enterprise. https://www.rijksoverheid.nl/documenten/publicaties/2021/02/12/google-workspace-dpia-for-dutch-dpa

Netherlands Court of Audit. (2025, January). Dutch central government in the cloud - report - netherlands court of audit [Last Modified: 2025-01-22T09:56 Publisher: Algemene Rekenkamer]. Retrieved March 3, 2025, from https://english.rekenkamer.nl/publications/reports/2025/01/15/dutch-central-government-in-the-cloud

OWASP. (2024). *Password storage cheat sheet.* https://github.com/OWASP/CheatSheetSeries/blob/083ec3d453cd157b92529b55ab9d733c55c7a81d/cheatsheets/Password_Storage_Cheat_Sheet.md

Puglierin, J., Varvelli, A., & Zerka, P. (2025). Transatlantic twilight: European public opinion and the long shadow of trump | ECFR. https://ecfr.eu/wp-content/uploads/2025/02/Transatlantic-twilight-European-public-opinion-and-the-long-shadow-of-Trump-v7.pdf

Rescorla, E. (2018, August). *The transport layer security (TLS) protocol version 1.3* (RFC8446). RFC Editor. https://doi.org/10.17487/RFC8446

Schildt, H., Förster, S., Hoffmann, B., Oppelt, J., & Welticke, J. (2023). *IT-grundschutz-kompendium* (6. Edition). Bundesamt für Sicherheit in der Informationstechnik.

Schryen, G., & Kadura, R. (2009). Open source vs. closed source software: Towards measuring security. *Proceedings of the 2009 ACM symposium on Applied Computing*, 2016–2023. https://doi.org/10.1145/1529282.1529731

Smith, J. C. (2022, April 30). Effective security by obscurity. https://doi.org/10.48550/arXiv.2205.01547

Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017). The first collision for full SHA-1 [Series Title: Lecture Notes in Computer Science]. In J. Katz & H. Shacham (Eds.), *Advances in cryptology – CRYPTO*

2017 (pp. 570–596, Vol. 10401). Springer International Publishing. https://doi.org/10.1007/978-3-319-63688-7_19

Teixeira, M. (2024). *Checkmarx advisory | cve-2024-4067 / regular expression denial of service (redos) in micromatch*. https://advisory.checkmarx.net/advisory/CVE-2024-4067/

Temoshok, D. (2024). *Digital identity guidelines: Authentication and authenticator management* (NIST SP 800-63B-4 2pd). National Institute of Standards and Technology. Gaithersburg, MD. https://doi.org/10.6028/NIST.SP.800-63B-4.2pd

Thompson, C. (2016, December 16). *Perils of the default bcrypt cost factor* [Medium]. Retrieved March 15, 2025, from https://labs.clio.com/bcrypt-cost-factor-4ca0a9b03966

Traefik Labs. (2022, May 5). *Rate limiting: What it is & why it matters | traefik labs* [Run APIs easily. anywhere. | traefik labs]. Retrieved February 23, 2025, from https://traefik.io/glossary/rate-limiting-what-it-is-and-why-it-matters/

Wiemer, F., & Zimmermann, R. (2014). High-speed implementation of bcrypt password search using special-purpose hardware. *2014 International Conference on ReConFigurable Computing and FPGAs (ReConFig14)*, 1–6. https://doi.org/10.1109/reconfig.2014.7032529