Single Sign-On Protocol Vulnerabilities: A Survey and How to Classify Them

Yanis Merabet, Lounas Ben Medjeber, Xiang Jiahui, Osman Salem, and Ahmed Mehaoua

Centre Borelli UMR 9010
Université Paris Cité
Paris, France
{firstname.lastname}@etu.u-paris.fr (Yanis, Lounas),
{firstname.lastname}@u-paris.fr (Other authors)

ABSTRACT

Single sign on (SSO) mechanisms have become essential in modern digital environments to simplify access to services while strengthening security. This article provides an in depth audit of several major SSO protocols OpenID, SAML, CAS, and WS-Federation by analysing their historical vulnerabilities. Using data from the CVE database, CVSS scores, and EPSS percentages, the study assesses the severity and likelihood of exploitation of the vulnerabilities identified between 2007 and 2025. The resulting visual and statistical analysis highlights the security trends specific to each protocol and brings forward key periods of heightened risk. To summarise these results, a star rating has been introduced, cross-referencing the levels of criticality (CVSS) and probability of exploitation (EPSS) for each protocol. This combined approach enables a clearer comparison of protocol robustness over time and supports informed decisions on protocol adoption and maintenance in real-world environments.

Keywords

Single Sign On, Authentication, SAML, OpenID, Cas, WS-Federation, Vulnerabilities, CLassification, CVSS, EPSS

1 Introduction

In today's hyperconnected digital landscape, authentication has become a foundational element of cybersecurity. It ensures that only authorized individuals gain access to digital resources, acting as a critical safeguard against data breaches and malicious activity. With the proliferation of online services and interconnected platforms, the demand for authentication mechanisms that are both robust and user friendly has never been greater.

Traditional authentication methods typically involving usernames and passwords are increasingly perceived as cumbersome and insecure. Users are often required to enter again their credentials for each service they use, which not only affects usability but also leads to poor password practices, such as reuse or weak password creation. These limitations expose systems to a wide range of vulnerabilities.

To address these challenges, the Single Sign-On (SSO) paradigm was introduced. SSO enables users to authenticate once and gain access to multiple applications or services without having to log in repeatedly. This streamlines the user experience, reduces friction, and improves overall security by centralizing identity management.

In this article, we take a closer look at four of the most widely implemented SSO protocols: Security Assertion Markup Language (SAML), OpenID Connect (OIDC), Central Authentication Service (CAS), and Web Services Federation (WS-Federation). Each of these protocols brings its own architectural approach, use cases, and security mechanisms, offering distinct advantages and trade offs.

To evaluate their security posture, we conducted a comprehensive vulnerability audit using data from the Common Vulnerabilities and Exposures (CVE) database. This analysis allowed us to identify the types and frequency of security flaws that have historically affected these protocols.

In order to provide a meaningful assessment, we built a classification model that takes into account not only the severity of each vulnerability—as measured by the Common Vulnerability Scoring System (CVSS) but also its likelihood of exploitation in real-world conditions, estimated through the Exploit Prediction Scoring System (EPSS).

By combining these two perspectives, our methodology offers a more nuanced view of risk, moving beyond theoretical severity to address the practical threat each vulnerability may represent.

Through this multi layered analysis, the article aims to offer readers a clearer and detailed understanding of how these authentication protocols function, the kinds of risks they face, and how organizations can make informed decisions about securing identity in federated environments [1] [2].

2 Related Work

Over the past two decades, Single Sign On (SSO) systems have been the subject of numerous studies seeking to balance ease of access and robust security. Subsequent work has focused on the analysis and comparison of SSO architectures and their attack surfaces. Mainka et al [8] proposed an explicit classification of SSO protocols according to their architecture (decentralized or monolithic) and their membership of the OAuth family. This typology enables vulnerability analysis to be structured according to common protocol characteristics. Alaca and van Oorschot [9], proposed frameworks to evaluate protocols like SAML, OpenID Connect, and CAS, using dimensions such as usability, deployment complexity, and security posture. Several studies have also tackled the security of these protocols through formal or empirical approaches. For instance, Fett et al. [5] analyzed OpenID Connect and identified structural weaknesses that could lead to authentication bypasses, while Hartl and Šerek [4] examined the SAML 2.0 browser profile to highlight risks related to assertion handling and message binding.

Despite this growing literature, few works have attempted to quantify both the severity and the exploitability of SSO related vulnerabilities over time. Our contribution fills this gap by combining CVSS and EPSS metrics to deliver a data driven, longitudinal risk classification across four major SSO protocols, offering a new lens to assess their evolving security landscape.

3 SINGLE SIGN ON

3.1 Definition

SSO is an identity management technique that allows users to log in once and gain seamless access to multiple resources or services without needing to re-authenticate for each one. These services can belong to the same organisation or be distributed across different trusted domains. This method streamlines the login process, reducing friction and repetitive actions for the user. In addition to enhancing usability, SSO also contributes to security by limiting the number of password entries and thereby reducing the surface for credential theft or phishing attempts. By centralising authentication, it also makes it easier for administrators to enforce consistent security policies. [3].

3.2 SSO architecture

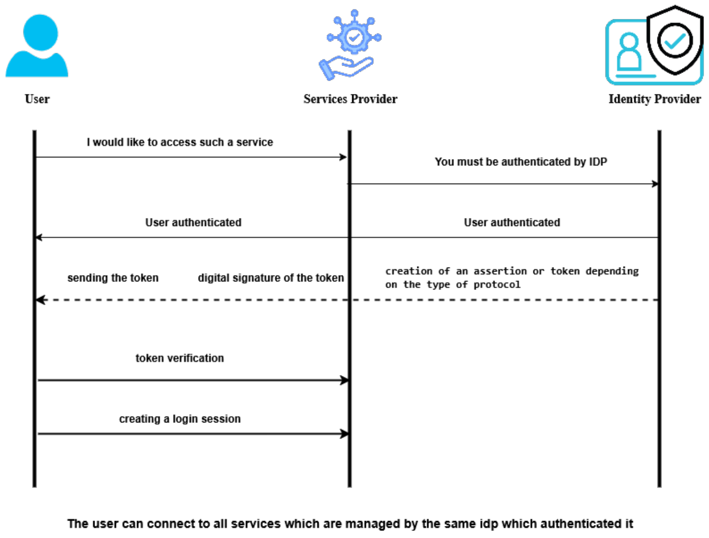
User
Services Provider
Identity Provider
I would like to access such a service
You must be authenticated by IDP
User authenticated
User authenticated
sending the token
digital signature of the token
creation of an assertion or token depending on the type of protocol
token verification
creating a login session
The user can connect to all services which are managed by the same idp which authenticated it

Figure 1: Typical SSO system architecture

SSO authentication simplifies access by allowing users to log in once and gain access to multiple services. The process starts when a user requests access to a service managed by an SP. This request is forwarded to an IDP, which validates the user's identity, often through methods like password or multi-factor authentication.

Once authenticated, the IDP creates a signed token or assertion, which is sent back to the SP via the user's browser. The Service Provider verifies the token's authenticity by checking the signature, ensuring it came from a trusted IDP. If the token is valid, the SP creates a session for the user, granting access to the requested service and potentially other services within the provider's ecosystem.

SSO streamlines the authentication process, reducing the need for multiple logins while maintaining security through token verification and centralized authentication.

3.3 Major SSO protocols

Below we present the most widely used protocols that implement the SSO mechanism in modern authentication systems.

- **SAML (Security Assertion Markup Language)** : SAML is an open standard based on XML that enables the secure exchange of authentication and authorization data between an IdP and a SP. Its main objective is to allow users to authenticate once and then access multiple independent services without having to log in again, a key principle of SSO.

  In practice, the IdP generates a signed assertion containing the user's identity and access rights, which is then transmitted to the SP. This mechanism eliminates the need for the Service Provider to manage user credentials directly, enhancing both security and usability. SAML is especially prevalent in web-based environments, where it enables smooth authentication integration across different organizations or domains, often in B2B or academic contexts. [4].

- **OIDC (OpenID Connect)** : OpenID Connect is a modern authentication protocol built on top of the OAuth 2.0 framework. It adds a standardized identity layer to OAuth, enabling applications not only to authorize access to resources, but also to verify the identity of the user. Through this mechanism, a user can authenticate with an IdP and then access multiple applications or services without needing to re enter credentials each time, a core feature of SSO

  OpenID Connect relies on the use of ID tokens, which are lightweight and secure data structures that contain identity information about the user. These tokens are transmitted from the IdP to the Service Provider after successful authentication, allowing the user to be recognized across different platforms.

  Thanks to its flexibility and simplicity, OIDC is widely adopted in web and mobile environments. It supports a broad range of use cases, from consumer facing platforms to enterprise level integrations. Its compatibility with RESTful APIs and support for modern web technologies make it a preferred choice for developers seeking to integrate secure and user friendly authentication [5].

- **CAS (Central Authentication Service)** : Central Authentication Service is an SSO protocol designed to allow a user to authenticate once and then access multiple applications without having to log in again. Initially developed by Yale University, it is widely used in academic institutions. CAS is based on a ticketing mechanism: after successful authentication, the CAS server issues a Service Ticket (ST), which the target application validates to grant access. This system guarantees that the user's identification information is never exposed to applications [6].
- **WS-Federation (Web Services Federation)** : WS-Federation is an authentication protocol that enables identity sharing across applications in different domains or organizations. After a user authenticates with an IdP, the IdP generates a security token, which is used to access services from various SP. This allows users to seamlessly access resources across multiple systems with a single login.

  Commonly used in Microsoft environments, such as Active Directory Federation Services (ADFS), WS-Federation facilitates Single Sign On (SSO) by enabling secure authentication across different platforms and simplifying user management. [7].

3.4 SSO protocol solutions

To better illustrate the relationship between existing products and the protocols they integrate, we present below a summary table showing the affiliation of each solution with the corresponding protocol [8] [9].

Protocole	**Produit**
OpenID Connect	Google Identity Platform Auth0 Okta
SAML	Shibboleth keycloak
WS-Federation	Active Directory Federation Services Azure Active Directory
CAS	Apereo CAS Jasig CAS

Table 1: Single Sign On protocols and their associated products

4 vulnerability attack analyses

4.1 Vulnerability over protocols

The SAML, OpenID Connect, CAS and WS-Federation protocols play a fundamental role in the implementation of SSO mechanisms, which facilitate centralised user authentication. However, despite their importance, these protocols are not infallible. Vulnerabilities can arise, whether as a result of errors in their design, flaws in their implementation, or misconfigurations in the environments where they are deployed.

Each time one of these vulnerabilities is identified, it is documented in the public CVE (Common Vulnerabilities and Exposures) database, which assigns each vulnerability a unique identifier accompanied by a technical description. This database is a valuable resource for researchers and cybersecurity professionals, as it enables them to track the evolution of vulnerabilities over time.

The figure below shows an annual breakdown of the number of vulnerabilities discovered for each of these SSO protocols, providing an overview of their evolution and security trends.

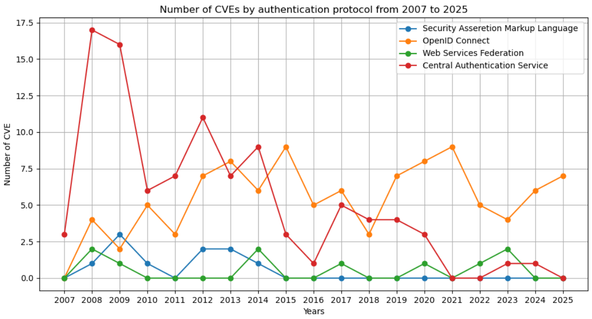

Figure 2: Number of CVEs by authentication protocol from 2007 to 2025

This graph shows that the SAML protocol has the highest number of vulnerabilities over the years, with a clear increase from 2015, peaking between 2020 and 2024. This is mainly due to its high adoption in large organisations and the complexity of its implementation, which increases the risk of misconfiguration.

OpenID also shows an irregular but significant rise, with recent peaks in 2023 and 2024. This trend seems to be linked to the growing popularity of its connect version (OIDC), which, if poorly integrated, can introduce vulnerabilities such as incorrect token verification or insecure redirections.

As for CAS, we have seen a gradual rise in vulnerabilities since 2018, no doubt reflecting wider adoption in academic and open source environments, where security practices vary widely.

Finally, WS-Federation remains the most stable and the least affected, which can probably be explained by its more limited adoption and less exposed attack surface.

4.2 Vulnerability over products

In order to better understand the real level of exposure to threats in the field, we conducted a detailed analysis of vulnerabilities discovered in a selection of widely used products that implement the SSO protocols discussed earlier. These products, often deployed in enterprise environments, represent concrete implementations where security flaws can have direct consequences. By examining the vulnerabilities reported for each of these solutions over time each identified in the CVE (Common Vulnerabilities and Exposures) database we gain insight not only into the frequency of security issues but also into recurring weaknesses or patterns. The graph below illustrates the number of CVEs associated with each product across multiple years, helping us trace the historical evolution of security concerns and clearly highlight which solutions have been most impacted. This longitudinal view provides a valuable perspective on the resilience of different implementations and their ability to respond to emerging threats.

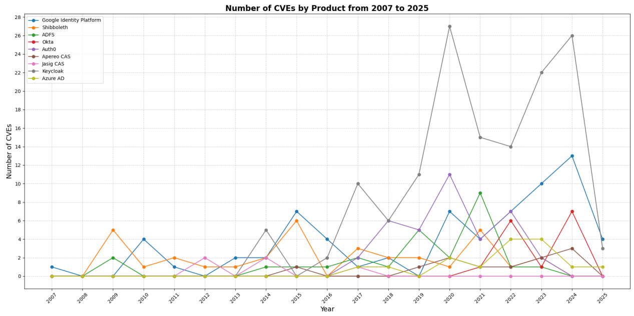

Figure 3: Number of CVEs by authentication product from 2007 to 2025

The graph showing the evolution of vulnerabilities between 2007 and 2025 shows us that the most popular and complex identity platforms, such as Keycloak and Google Identity Platform, have the highest number of vulnerabilities, particularly in recent years. This trend can be explained by their massive adoption, their technical complexity and the constant attention they receive from the cybersecurity community.

We also note that Keycloak, with a notable peak in 2020, reflects this dynamic characteristic of widely used open source projects. Azure AD, meanwhile, is relatively stable, reflecting Microsoft's rigorous internal controls. Solutions such as Auth0 and Okta show occasional peaks linked to their functional expansion.

Conversely, we note that more traditional or academic tools such as Apereo CAS, Jasig CAS, Shibboleth and ADFS show little or no significant fluctuations, partly due to their more restricted functional scope or lower exposure.

This overview shows that the discovery of vulnerabilities is often a reflection of heavy use, and not necessarily of a lack of security, underlining the importance of continuous monitoring and regular updates.

5 classification model

As part of our audit of authentication protocols, we used three key criteria to establish a vulnerability classification model. These criteria are the CVSS score, which is a standardised scoring system that evaluates the severity of vulnerabilities. It assigns each vulnerability a numerical score ranging from 0 to 10, where 0 means no severity and 10 indicates a critical vulnerability. We also considered the EPSS percentile, an indicator of the potential for a vulnerability to be exploited within a given timeframe. It represents a relative value indicating where a vulnerability stands in relation to all the others, and provides a statistical estimate of the probability that it will be exploited within 30 days of its publication. Finally, the year criterion is used to analyse the temporal distribution of vulnerabilities. It helps to identify critical periods when certain vulnerabilities have been particularly numerous, and to observe trends in the improvement or deterioration of security over time. This criterion highlights changes in the resilience of protocols in the face of emerging threats.

By cross referencing the criteria used for the analysis, we generated four graphs, each corresponding to one of the protocols studied. They represent the dangerousness and criticality of the vulnerabilities over the years.

5.1 Classification of SAML

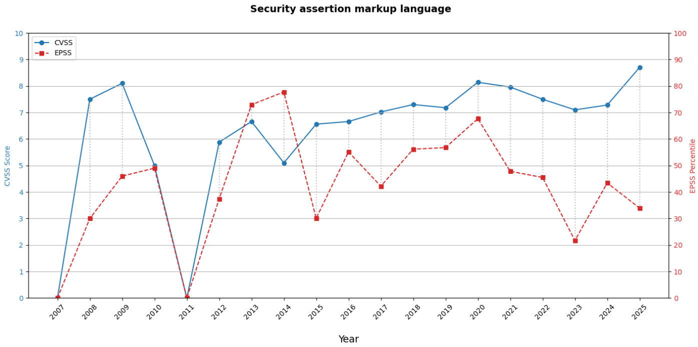

Figure 4: Security assertion markup language

The SAML protocol has a delicate security profile: its vulnerabilities are often critical, but their exploitation depends heavily on the technical and organisational context. For example, in 2013, with a CVSS vulnerability score of 6.7, the percentile EPSS climbed to almost 73%, a sign of rapid exploits in popular deployments; conversely, in 2015, despite a CVSS of 6.6, the percentile EPSS only reached around 30%, reflecting more limited distribution and patches deployed before massive attacks became widespread. Similarly, the peak in 2020 (CVSS at 8.2, percentile EPSS at 68%) shows a strong correlation between severity and the probability of exploitation, while in 2023, the CVSS remains high (7.1) but the percentile EPSS falls to around 22%, revealing that, despite the technical severity, the conditions for exploitation have been contained (configuration required, less interest from attackers or rapid responses, patching, etc.). These fluctuations clearly illustrate that, for SAML, a high CVSS score does not guarantee an immediate risk of attack: only continuous monitoring and proactive updates can reduce the actual exploitation window.

5.2 Classification of OpenID

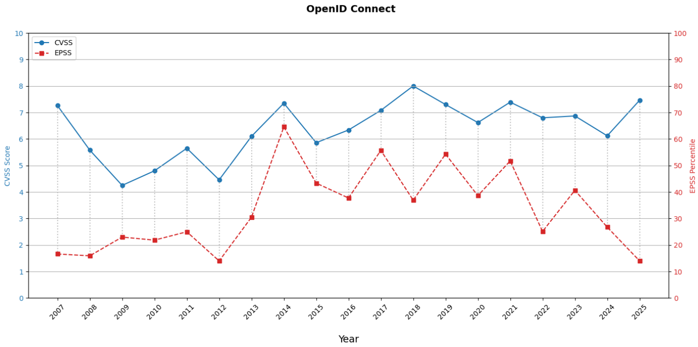

Figure 5: OpenID Connect

We have seen that the OpenID Connect protocol has undergone a particularly delicate security evolution over the years. Looking at the graph, we see that the CVSS scores remain predominantly high, reflecting vulnerabilities that are often critical or technically important. However, by cross-referencing this data with the percentile EPSS percentiles, we can see that the actual exploitability of these vulnerabilities varies greatly from year to year. For example, in 2014, we observed a significant spike in both the CVSS score (7.4) and the percentile EPSS (almost 70%), indicating a critical vulnerability that was actually exposed, probably facilitated by the widespread use of OpenID in third party authentication services without sufficient controls. Conversely, in 2018, even though the CVSS reaches a very worrying level (8.0), the percentile EPSS remains moderate (around 37%), which tells us that this vulnerability, although serious, seems less likely to be exploited in practice, no doubt due to technical barriers or restricted use of the functionality concerned. By following these trends, we can clearly see that not all critical vulnerabilities are immediate threats, but that they represent a latent risk that deserves constant attention. That's why we need to remain vigilant, because vulnerabilities that are not considered exploitable today can quickly become exploitable with the evolution of uses, attack vectors or bad configurationsin the actual deployment of the protocol.

5.3 Classification of WS Federation

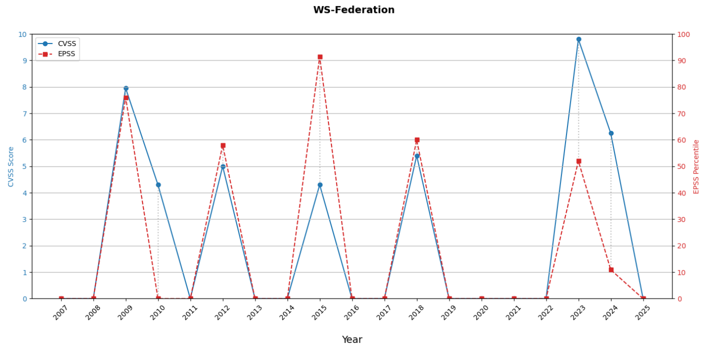

Figure 6: Web Services Federation

We have seen that the WS-Federation protocol has a very irregular security trajectory, marked by long periods of apparent calm interspersed with sudden spikes in critical vulnerabilities. From the graph, we can see that certain years, such as 2009, 2015 and especially 2023, are characterised by very high CVSS scores (around 8, 4.3 and an extreme peak of 9.8 respectively), reflecting potentially very serious technical flaws. What draws our attention is that in several of these cases, the EPSS percentile is also very high notably in 2015 (EPSS at 94%), suggesting that the vulnerabilities identified were not only critical, but also actively exploitable or at least easy to exploit. In 2023, we see a striking contrast: a CVSS score of 9.8, i.e. practically the maximum, but a more moderate EPSS (around 53%), which leads us to believe that despite the technical severity, exploitation probably depended on specific conditions or rarely used configurations. On the other hand, we note many years in which no risk is reported (CVSS and EPSS at zero), which may reflect a period of stability or a lack of visibility of the vulnerabilities actually present. These oscillations reinforce our impression that WS-Federation, although discreet, is not immune to major security incidents. We must therefore remain vigilant, as these vulnerabilities can reappear suddenly, with a major impact, especially in environments that continue to rely on this protocol for identity federation. impact, especially in environments that continue to rely on this protocol for identity federation.

5.4 Classification of CAS

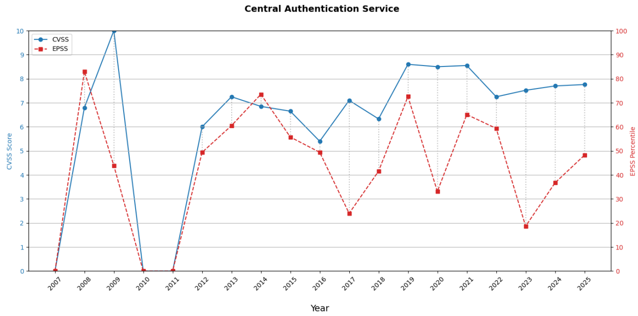

Figure 7: Central Authentication Service

We have observed that the Central Authentication Service (CAS) protocol presents an overall high and relatively constant level of risk, with critical peaks reflecting major vulnerabilities. In particular, the graph reveals an exceptional flaw in 2009, when the CVSS score reached a maximum of 10, accompanied by a still moderate EPSS (around 43%), suggesting a theoretically catastrophic vulnerability whose exploitation seems to have been slowed down by technical or contextual factors. On the other hand, in 2008 and 2014, we found significant agreement between the CVSS and EPSS scores: in 2008, for example, with a CVSS of 6.9 and an EPSS of over 80%, which suggests that the flaw was widely accessible to attackers. Over the years, we see a persistence of critical vulnerabilities: from 2013 to 2025, the CVSS score very often remains above 6, even reaching 8.7 in 2019, which reflects a lack of structural robustness or poor patch lifecycle management. However, the variations in the EPSS, for example the drop in 2023 despite a CVSS of around 7.5, indicate that the potential for exploitation fluctuates greatly depending on the technological context or active attack campaigns. We must therefore conclude that, although the CAS protocol has a relatively well identified security framework critical vulnerabilities persist and can become seriously exploitable as soon as an exploitation vector or opportunity opportunity for exploitation arises. This requires continuous monitoring and rigorous updating to reduce exposure.

5.5 comparison table

To synthesize the insights drawn from our vulnerability audit, we present a comparison table that evaluates each SSO protocol using two key indicators: the CVSS score and the EPSS percentile. This dual axis evaluation allows us not only to gauge the technical severity of the vulnerabilities (via CVSS), but also to estimate their actual risk of exploitation in the wild (via EPSS).

Each protocol is assigned a star-based rating for both criteria. This scale ranges from one star, indicating very low risk or severity, to five stars, reflecting a critical level. An intermediate number of stars corresponds to increasing levels of concern, offering a quick visual representation of how each protocol stands in terms of both danger and exploitability. This approach simplifies complex technical data into an accessible format, making it easier to communicate findings to both technical and non-technical stakeholders. It also helps highlight which protocols may require immediate attention or mitigation efforts.

Protocols	**CVSS**	**Percentile EPSS**
SAML	★ ★ ★ ★	★ ★ ★
OpenID	★ ★ ★ ★	★ ★
WS-F	★ ★	★
CAS	★ ★ ★ ★	★ ★ ★

Table 2: Comparison of protocols according to two criteria

We would like to make it clear that:

★	Very Low
★ ★	Low
★ ★ ★	Moderate
★ ★ ★ ★	High
★ ★ ★ ★ ★	Critical

6 Conclusion and Future Work

Through our in-depth analysis, we have constructed a comprehensive picture of the vulnerabilities affecting the most widely used Single Sign-On protocols. Rather than limiting ourselves to traditional severity ratings, we adopted a cross referenced methodology that combines CVSS scores with EPSS probabilities. This dual layered approach enabled us to move beyond theoretical severity and focus on the practical likelihood of exploitation, offering a richer and more grounded perspective on security risks.

What emerged from this analysis is that each SSO protocol carries its own distinct risk profile, shaped not only by its technical design but also by how and where it is deployed. Popular protocols like SAML, due to their widespread use, naturally draw more attention from attackers, potentially increasing the frequency of exploit attempts. On the other hand, less common or more technically intricate protocols might expose systems to niche or implementation specific vulnerabilities that are harder to detect but no less dangerous.

By identifying these risk patterns, we aim to contribute meaningful insights to the broader conversation around federated authentication security. Our hope is that this work helps bridge the gap between high level vulnerability databases and the nuanced, real world considerations that security practitioners face when selecting, configuring, or maintaining authentication systems.

Moreover, this research reinforces our conviction that security is not a one time evaluation it must be a continuous process. The threat landscape is dynamic: new attack techniques emerge, protocols evolve, and usage contexts shift. As such, periodic reassessment and constant monitoring are essential to ensure that what was secure yesterday remains secure tomorrow.

Looking ahead, we intend to expand our framework to encompass emerging authentication standards, recognizing that innovation often introduces new complexities and risks. We also plan to refine our risk classification model by integrating additional context aware criteria potentially including deployment scale, threat actor interest, and historical exploit trends to improve the precision of our assessments.

Finally, we envision the development of an interactive, user friendly visualization tool that will make this information more actionable for decision-makers. By providing clear, intuitive representations of protocol vulnerabilities and associated risks, we hope to support organizations in making informed, strategic decisions to better protect their systems, users, and data.

References

[1] Y. R. Avuthu, "Federated identity and single sign-on (sso): Balancing security and usability in cloud iam implementations," *The Journal of Scientific and Engineering Research*, vol. 6, pp. 239–247, 01 2019.

[2] A. Josang, M. AlZomai, and S. Suriadi, "Usability and privacy in identity management architectures," in *ACSW Frontiers 2007: Proceedings of 5th Australasian symposium on grid computing and e-research, 5th australasian information security workshop (privacy enhancing technologies), and Australasian workshop on health knowledge management and discovery.* Australian Computer Society, 2007, pp. 143–152.

[3] T. Bazaz and A. Khalique, "A review on single sign on enabling technologies and protocols," vol. 151, pp. 18–25.

[4] Z. Hartl and A. erek, "Towards automated formal security analysis of saml v2. 0 web browser sso standard–the post/artifact use case," *arXiv preprint arXiv:2403.11859*, 2024.

[5] D. Fett, R. Küsters, and G. Schmitz, "The web sso standard openid connect: In-depth formal security analysis and security guidelines," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 2017, pp. 189–202.

[6] A. Amarudin, "Implementation of cas server as authentication protocol on single sign-on (sso) network with php programming," 12 2014.

[7] M. Ates, C. Gravier, J. Lardon, J. Fayolle, and B. Sauviac, "Interoperability between heterogeneous federation architectures: Illustration with saml and ws-federation," in *2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System.* IEEE, 2007, pp. 1063–1070.

[8] C. Mainka, V. Mladenov, T. Guenther, and J. Schwenk, "Automatic recognition, processing and attacking of single sign-on protocols with burp suite," in *Open Identity Summit 2015.* Gesellschaft für Informatik eV, 2015, pp. 117–131.

[9] F. Alaca and P. C. V. Oorschot, "Comparative analysis and framework evaluating web single sign-on systems," *ACM Computing Surveys (CSUR)*, vol. 53, no. 5, pp. 1–34, 2020.