Information & Application Security

White Paper

February 15, 2019

Table of Contents

Introduction

This White Paper provides an overview of the security measures of Semantic Web Company undertaken on an organizational and product level. It shall support decision makers during their vendor and software evaluation process.

Semantic Web Company, headquartered in Austria, with a branch in the UK, is the leading provider of graph-based knowledge technologies. It is also the vendor of PoolParty Semantic Suite, the industry's most complete semantic middleware platform on the global market. PoolParty Semantic Suite uses innovative means to help organizations build and manage enterprise knowledge graphs as a basis for various AI applications. Additionally, a certified partner network complements their technology services.

According to KMWorld, Semantic Web Company has been named as a company that matters in Knowledge Management. With Credit Suisse, Roche, Philips and the World Bank among their customer base of Global 2000 companies, Semantic Web Company is continually helping customers scale their AI strategy and extend it to fit their organizations.

Implementing Semantic AI in enterprise information management systems is an innovative and complex encounter. As we work with large, distributed and heterogeneous data sources, security questions are of special concern. PoolParty Semantic Suite is enterprise-ready and ISO 27001:2013 certified. We embrace a transparent approach towards our customers in regards to internal security processes, applied technologies and third-party vendor relations.

You will find further resources at the end of the White Paper. Alternatively, you can reach out to our information security team at: security@semantic-web.com.

Andreas Koller	**Johannes Trippl**	**Michael Scharitzer**
CIO	Head of System Operations	Information Security Management

Our approach to information security

The Semantic Web Company was established in 2004 and is a distinguished expert organization in the fields of semantic AI technologies and standards-based semantic metadata management solutions. Among our customers are government organizations and global organizations across multiple industries, such as: financial services, e-commerce, pharmaceutical industry and media & publishing. These companies have complex data models at the core of their business and operate with sensitive data repositories ranging from customer, product and provenance data to intellectual property.

PoolParty Semantic Suite is a semantic middleware, which helps to enrich heterogeneous data sources with metadata,making it actionable across multiple platforms, while taking context dependency into account. Our technology solutions are deeply integrated in enterprise information architectures. With the growing demands in improved search, recommendation and dynamic content publishing services, the semantic capabilities of smart data platforms have become increasingly important to organizations.

Applying Semantic AI is an exciting journey. It might start out as an innovation project, but usually with a high awareness of its strategic importance for the whole enterprise. PoolParty customers use the semantic software in various ways and many different IT-systems can gradually merge into a cognitive computing platform. PoolParty is a technology suite based on W3C standards, which is thoroughly designed to stay integrable and state-of-the-art in the long term.

This has severe implications for our information security approach. We conform to security standards that fit with the compliance expectations of Fortune 500 companies. At the same time, we are aware that IT regulations shouldn't be too restrictive in a rapidly changing environment. Balancing security requirements with enough alternative choices for our customers is a core principle for our information security management committee.

Cybercrime is an actual threat that can not be solely prevented on a technical level. Building and sustaining security awareness and limiting operational risk across the whole organization is key. The top management of the Semantic Web Company is actively driving information security measurements to protect critical assets and considers security questions at every relevant touchpoint.

Corporate Security

The Semantic Web Company is a globally acting organization in a highly networked business and knowledge ecosystem. An information security expert team sponsored by top management ensures that all stakeholders comply with the code of conduct and consider potential risks in their daily work carefully. As the vendor of a semantic data management platform for data intense industries, fulfilling high security standards is a critical prerequisite.

Background checks

Depending on the professional role of our employees, background checks on different intensity levels are performed. 80% of our employees are recruited internationally and must go through severe national security screening from the Austrian government. Many of our consultants participate in international government projects, where security clearance is mandatory as well.

A partner certification program ensures that strategic partners for PoolParty Semantic Suite share the same quality standards along the whole value chain.

Security Awareness Program

Security rules, regulations and guidelines affect all organizational processes and departments. A strong security culture depends on every employee and needs to be continuously screened and adapted. Every new employee gets a dedicated security onboarding. Regular updates of the corporate security framework are shared by the information management security committee. Additionally, every department has a security commissary who ensures that teams act accordingly to regulatory compliance.

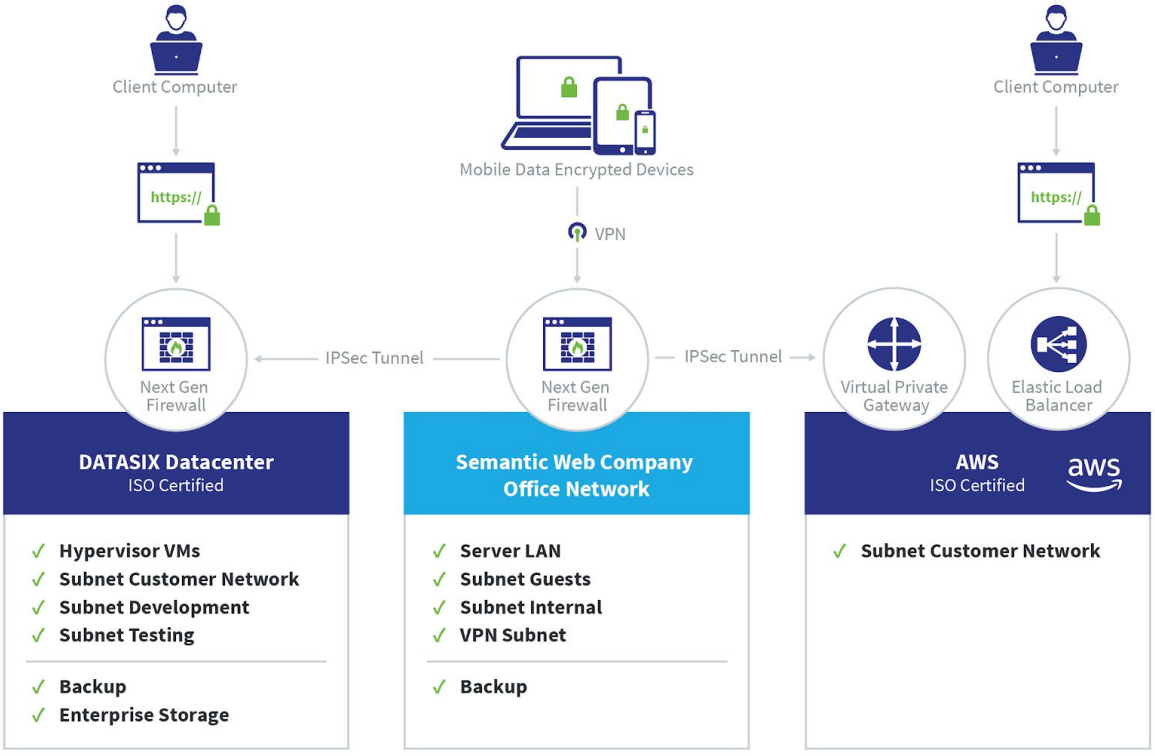
Data privacy

Since the GDPR (General Data Protection Regulation) has been taken in effect since May 2018, there has been extensive mandatory rule set for data privacy to follow. European companies have to document transparently which personal data is collected and how this data is processed. A business process inventory provides the underlying foundation to keep track of potential data breaches, which also strengthens the internal security awareness as a side effect. Having an overview of data flows is an important strategic asset for keeping the information security management program up-to-date. Non-EU companies benefit from legally binding data privacy regulations.

Network security

The rise of cloud tools, mobile devices and working virtually requires preventative measures to ensure improved data security. Only authorized users have access to the Semantic Web

Company's network. Remote access is strictly controlled with encryption and a strong password policy. Network traffic from and to the data center and AWS are secured via IPSec tunnels. At the data center, the network is segmented. Customers and various company departments, such as Infrastructure, Development and Testing, have their own subnet. A next generation firewall prevents a broad range of security threats and a monitoring system immediately notifies about malicious activity. By performing regular audits, potential security vulnerabilities are examined and considered in the strategic security roadmap. A multi-layered security approach is established for utmost protection of the organization's IT assets, while still granting enough flexibility to ensure an agile business environment.



Semantic Web Company Network Security Architecture

System level security

All virtual machines use an operating system that is hardened to reduce the surface of vulnerability. Every software package which is not needed by the server or the infrastructure is removed. In order to secure all systems, users only have a specified set of commands with elevated rights. The system operations team reviews every request for access to a server as well as needed software packages. Malware detection and file integrity checks are deployed on all servers.

System Recovery Mechanisms

All company data is backed up daily to a separate data storage server in a data center at a different geolocation. Separate local copies of data are backed up to a mobile storage in a secure and locked location. All backups are highly encrypted and protected from unwanted access. To keep recovery times short, snapshots are made for every virtual machine which are kept and backdated for seven days. Additional snapshots of data are made in the datacenter on an enterprise storage device. All data is kept for 50 days at the offsite backup location.

Physical Protection

Our major company servers are running in ISO 9001:2008 and ISO 27001:2005 certified data centers. The data centers are under permanent video surveillance. All rooms are monitored around the clock, seven days a week by a security firm. Checkpoints are used to log and record regular scrutiny. Entrance is only possible by passing an RFID and a biometric access system. The data center has a fire alarm system with inert gas. The Very Early Smoke Detection is very sensitive and is already alarmed, when there are only a few smoke particles in the air.

Risk Assessment and Audit Arrangements

The Information Security Management System (ISMS) of the Semantic Web Company is based upon the ISO/IEC 27001 standard. A risk assessment and risk treatment methodology has been established to address unwanted events in the most efficient way.

Product Security

PoolParty Semantic Suite is a semantic data integration platform that integrates with third-party enterprise information systems. The data processing capabilities and embeddedness in core business functionalities require high security measures along the whole software development lifecycle. PoolParty Semantic Suite provides two major releases every year. Our software engineering and systems delivery team follows the OWASP (Open Web Application Security Project) principles.

Cloud or on-premise installation

PoolParty Semantic Suite can be licensed as an on-premise installation or as cloud service. Depending on the IT strategy of our customers, various deployment scenarios are feasible. In regards to security, it depends on the customer's preferences which alternative is most suitable. For all deployment options, support packages are available.

Customers usually decide for an on-premise installation, when PoolParty is heavily integrated with third-party systems and a mature IT environment is in place.

Many companies run their software services in the AWS (Amazon Web Services) cloud and can easily add PoolParty Semantic Suite to it. In this case, the customer will be responsible for hosting and updating PoolParty Semantic Suite themselves.

Alternatively, Semantic Web Company can host and update PoolParty Semantic Suite in the AWS cloud or the DATASIX datacenter. Customers will always have the newest software version available. Shared security responsibilities with the cloud provider will be covered by Semantic Web Company, which significantly reduces the operational burden. The two datacenters provide equally high global infrastructure security, but provide different pricing models.

Secure Software Development Process

Our software development process lifecycle covers security issues stemming from customer security audits as well as regularly performed internal audits that are aligned with the OWASP Top 10 risks.

To increase security awareness and strengthen secure coding skills of the whole development team, regular trainings are organized. Peer-reviews of new features ensure high quality of code and mitigate the risks of security related issues during the design and development phase. Furthermore, legacy code and outdated external dependencies (i.e. libraries) are regularly scheduled for redesign and refactoring.

Third-party components are reviewed exhaustively before used in the core product, and are regularly scanned for common vulnerabilities. All security critical functionalities such as authentication, authorization and encryption are implemented based on well-established (standard or de-facto standard), proven and actively maintained frameworks.

Highly customizable security functionalities

PoolParty provides default configurations for all security critical functionalities that can be easily configured and extended for the customers' security policies. This includes the enforcement of secure user passwords, configurable session timeouts, and authentication methods.

New security configuration options and features such as password history, captcha support for locked users, or more recent encryption and hashing methods (such as Argon2) are regularly included in the release cycles to meet the requirements of different security policies.

Authentication

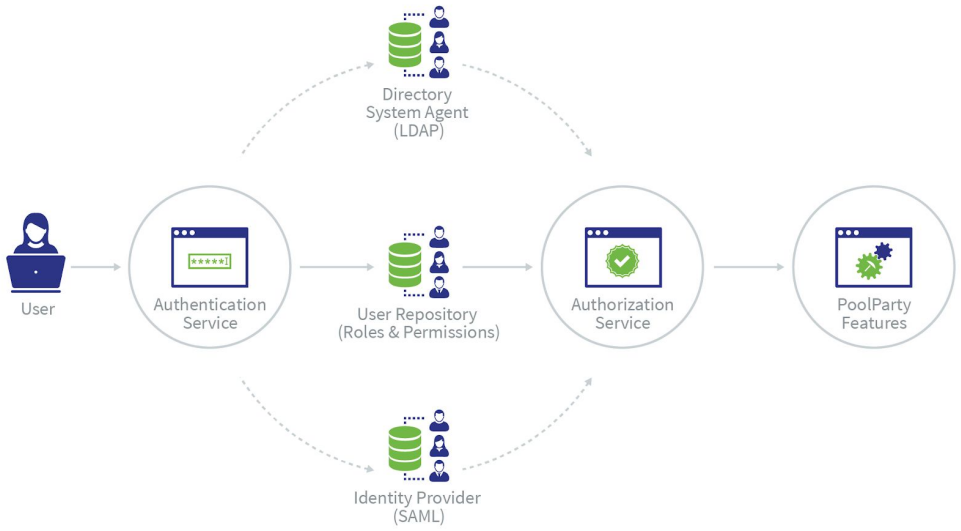
PoolParty Semantic Suite provides various ways to authenticate users.

The default authentication method is password based verification. Administrators can configure a strong password policy by enforcing a minimum password length,mixed cases, as well as digits and punctuation. Maximum login attempts and locking/unlocking options are also provided.

Some customers might prefer to use their existing authentication infrastructure, PoolParty Semantics Suite's authentication can be enforced alternatively via:

- Lightweight Directory Access Protocol (LDAP)
- Single-Sign-On (SSO) using Security Assertion Markup Language (SAML 2.0)

Server administrators can also restrict access from predefined IP addresses or domain names.



PoolParty Authentication and Authorization

Authorization

Authorization determines the rights to see, create and change data. The user management module of PoolParty Semantic Suite differentiates various roles. With release 7.0, the role-based system has been extended to include permission-based access control, allowing more flexible and fine grained access rights.

For example:

- Differentiate access rights via user roles superadmin, admin, use, API user, read-only user
- Set up appropriate access rights for different PoolParty projects by creating user groups
- Make projects publicly available by publishing them via the Linked Data front-end

Session management and secured API endpoints

Every successfully logged-in user is assigned a session that identifies the user and applies access control restrictions upon subsequent requests. Server administrators can easily change the session timeout to mitigate the risks of users leaving their workplaces while still logged in.

Every request to the provided Application Programming Interfaces (REST API) is secured by basic authentication. Secure endpoints must only be provided over HTTPS connections, which protects user credentials and guarantees integrity of sent data. This is fully supported by the PoolParty server and documented in detail in the system installation guide.

Monitoring and data recovery

System administrators set up different log configurations for the components of the product, which allows the monitoring of different events such as failed login attempts or current logged in users.

The project history allows the tracking of changes to project data, providing the means for manual inspection or custom implemented analysis. PoolParty 7.0 is able to configure the maximum number of failed login attempts stored for each user. The notification management system is able to configure alerts for different user actions that can be communicated over different channels.

In case of data corruption, automatically created snapshots enable the recovery of project data.

Further reading

Different stakeholders in your organization have various information needs when it comes to IT security. In this White Paper we provided you with a general overview of the security and risk management measures of the Semantic Web Company at a corporate and product level.

- Find more product-related security information at: https://help.poolparty.biz
- For customer references, please visit: https://www.poolparty.biz/customers/
- Explore our partner network at: https://www.poolparty.biz/partners/
- Our information security team is happy to answer your questions: security@semantic-web.com

Appendix A - PoolParty Semantic Suite Security in Detail

PoolParty Semantic Suite's security measures are based on proven state-of-the-art frameworks and standards. Based on the provided reference implementations, the available security features, measures and integrations are constantly updated and extended to guarantee system and data security. The following sections list the most important features and concepts applied in PoolParty Semantic Suite.

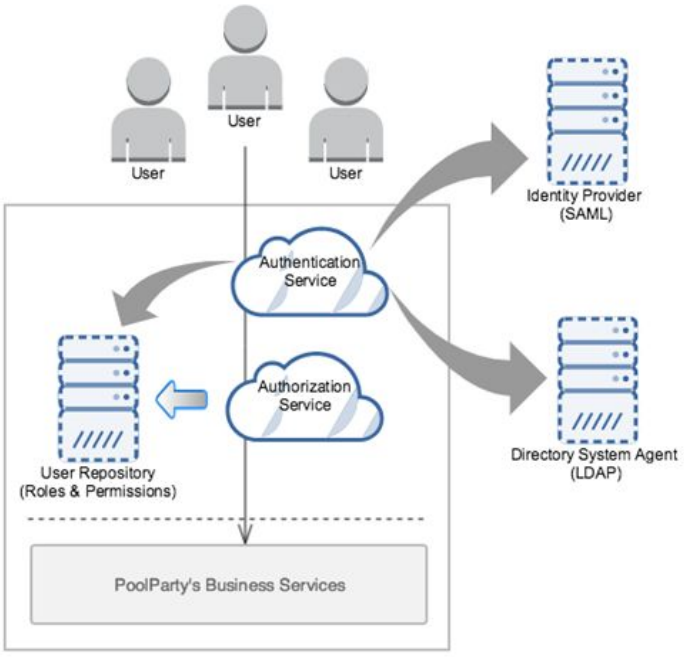
Authentication

> Authentication is the process of verifying the user's identity based on credentials (e.g., username and password).

Authentication in PoolParty Semantic Suite is based on the "Spring Security" framework [1], which provides a trustworthy default implementation and allows for a range of customizable security features.

The default implementation enforces:

- Form-based Authentication (session-based authentication) for the User Interface (e.g., PoolParty Thesaurus Manager),
- Basic Auth for API endpoints (this is additionally secured by the HTTPS protocol layer endorsed by PoolParty)



The implementation covers secure hashing of stored user passwords (using the Argon2 algorithm) and encoding of system credentials. The password strength can be easily configured [2], as well as the maximum number of authentication attempts before the user is disabled. For re-enabling, validation via captcha can be configured. Also, users may be forced to use a new password on change by keeping track of password history.

Other authentication methods (e.g. Digest Auth), protocols, and providers can be configured, allowing it to use LDAP [3], SAML [4], Active Directory, OAuth, OpenID and even custom implementations.

[1]https://spring.io/projects/spring-security (version 5)

[2]https://help.poolparty.biz/doc/administrator-guide/poolparty-installation/advanced-configuration/configure-poolparty-to-use-stronger-passwords

[3]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/poolparty-user-administration/setup-ldap-authentication-for-poolparty

[4]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/poolparty-user-administration/setup-saml-authentication-for-poolparty

Authorization

> Authorization describes the process of defining access rights to users or user groups.

PoolParty Semantic Suite supports role-based access control [1] where several user groups are able to manage data classification of a project, i.e.

- public information, that may be freely disclosed with the public;
- user groups, information that is only visible to users belonging to a specified group.

For each project, at least one user group has to be configured. Furthermore, since PoolParty 6.1, a Read-Only [1] role allows the user to see existing information without giving the user the opportunity to create or modify data in a project.

The Thesaurus Manager can be used to configure the access-controls for user accounts and groups.

Access levels for the Wiki front-end can also be configured server-wide [2], e.g., with ability to disable public access for the entire server.

[1]https://help.poolparty.biz/doc/user-guide-for-knowledge-engineers/basic-features/the-user-administration-overview/create-a-new-user-define-a-custom-uri/user-roles-in-poolparty

[2]https://help.poolparty.biz/doc/administrator-guide/poolparty-frontend-configuration/configuring-the-frontend/access-levels-for-the-frontend

Notes/Roadmap:

· Permission-based access control

Data Security and Secure Transmission

> Data security and secure transmission refers to the measures that are taken for protecting the stored data from malicious attacks, such as destructive forces or sniffing.

By default, all data used by PoolParty Thesaurus Manager is stored in the local file system (i.e., on the Server, see PoolParty data directory). This option ensures that:

- Thesaurus data is stored in an RDF database backed by the local file system
- Extraction data is stored only in a local Solr server

Sensitive data is always hashed (for passwords) or encrypted (for other sensitive data including API keys):

Since PoolParty 6.0, the Semantic Middleware Configurator [1] allows to manage the connection to third-party graph databases, search indices (e.g. ElasticSearch), and various integrations such as JIRA instances or visualizations. Third-party passwords (e.g., passwords of configured triple stores) managed by PoolParty are always encrypted using appropriate security frameworks (i.e. AES, supporting configurable key size (up-to 256-bit), salt and random initialization vector).

Data loss can be prevented by setting up an appropriate backup strategy. The recommended backup strategy [2] incorporates regular snapshots automatically made by PoolParty, mitigating the effects of hardware-related server incidents or malicious attacks.

For communications security over a computer network, SSL/TLS [3, 4] is highly recommended and can be configured in PoolParty's Tomcat. Through Tomcat's connector configuration, the entire communication with the PoolParty server can be encrypted using an SSL connector. A valid certificate from a trusted CA is required (you can get a free certificate from, e..g, "Let's Encrypt" [5]).

[1]https://help.poolparty.biz/doc/user-guide-for-knowledge-engineers/enterprise-features/semantic-middleware-configurator-overview

[2]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/recommended-backup-strategy-for-a-poolparty-server

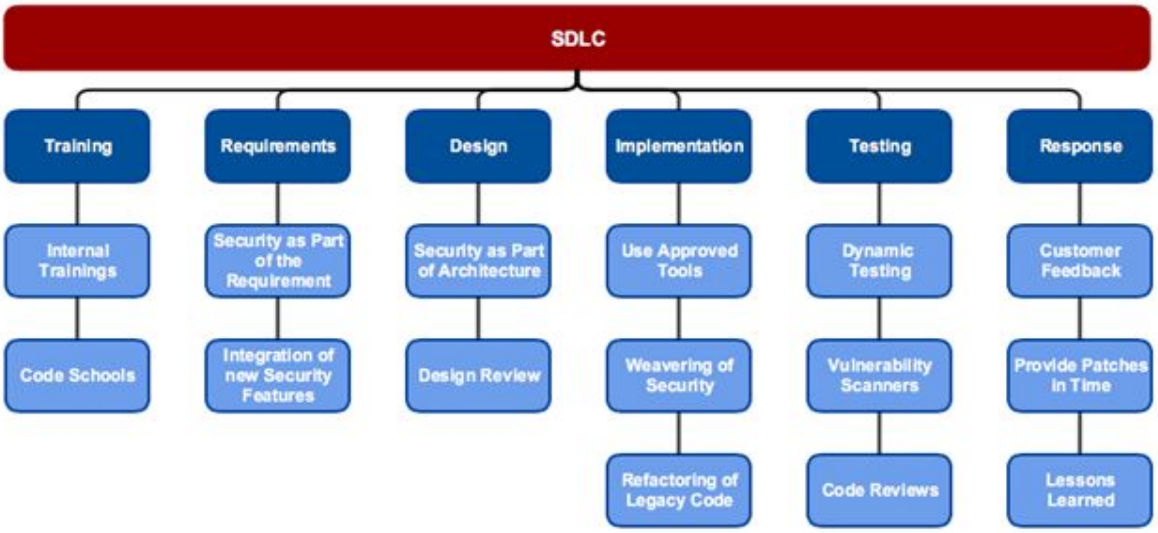
[3]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/security-configuration/provide-secure-backend-for-ppt-linux

[4]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/security-configuration/provide-secure-backend-for-ppt-windows

[5]https://letsencrypt.org/

Secure Software Development Cycle

In order to maintain and improve the security of the PoolParty Semantic Suite, the development is based on a secure software development life-cycle depicted in the following figure. The individual phases of the life-cycle are briefly explained in the following sections.



Training

The training component incorporates all activities to:

- Increase the awareness regarding the importance of security in the development phase by using internal trainings, and
- Train developers to write secure code by using code schools

The core part of this activity is the definition of expert roles and contact persons.

Customer feedback and lessons learned are reviewed and integrated in the training process.

Requirements

PoolParty Semantic Suite is a product that is constantly subject to further development and relies on continuous security improvements that are planned in the form of requirements/user stories.

- For new functional requirements, security concerns are formulated within the user stories such as access controls, etc.
- Security features are part of the PoolParty Semantic Suite roadmap and are based on customer demands or result from internal audits and risk analysis of the software (see Risk Analysis, OWASP Top 10).

Design

In the design phase, security related concerns are identified, analyzed and added to the requirements:

- Security is always treated as an integral part of the system architecture and as such part of every feature design (considering consequences of user stories regarding security aspects, carefully aligning with access control and other security aspects described in this document)
- Design suggestions are discussed within the team and reviewed by experts, always challenged by latest developments in the security domain (closely following latest findings and advances in the field)
- New requirements are written based on user stories that include access control descriptions

Implementation

In the implementation phase, the following guidelines are followed:

- Careful selection and evaluation of proven, secure and reliable tools and frameworks, adhering to (de-facto or formal) standards where possible
- If not available, functionality is implemented in common services that are reused and tested extensively
- Constantly updating libraries and dependencies to latest security patches (using a consistent version across all PoolParty Semantic Suite components)
- Continuous refactoring and maintenance to improve security and fix weak code as part of normal development cycle

Testing

Testing is an integral part of the development life-cycle during implementation, at the end of sprints and before release:

- Automated module tests (every feature developed is unit tested) during implementation and as part of continuous integration (using Jenkins build server)
- End-to-end tests performed by a dedicated testing team
- Using vulnerability scanners that are constantly improved by development, sysops and testing teams
- Exhaustive code reviews to identify remaining critical code

Roadmap:

- Evaluation and adoption of additional Source Code Analysis Tools (SAST [1] and DAST [5] tools), esp. [2] and [3].
- Training testers to perform in-house vulnerability testing

[1]https://www.owasp.org/index.php/Source_Code_Analysis_Tools

[2]https://www.owasp.org/index.php/OWASP_SonarQube_Project

[3]https://www.owasp.org/index.php/Category:OWASP_Orizon_Project

[4]https://www.owasp.org/index.php/Category:Vulnerability_Scanning_Tools

Response

In the response phase, all teams gather to collect and share insights, target critical issues, and transfer outcomes into upcoming training and requirement phases:

- Customer feedback and lessons learned are used for further training sessions and requirements
- Emerging security issues are identified and patched by prompt hotfix releases
- Close feedback loops between development and system operations to share relevant details from development that affect deployment, and information from sysops regarding newly discovered security threats in the field to check in development

Risk Analysis

Application security is tightly coupled to risk management, as security exploits and resulting damage can have severe impact on operations and even the business as a whole.

Therefore we follow the OWASP security guide and the risk analysis outlined below. It is important to note that exploitation strategies for several known security vulnerabilities are often combined in order to achieve more effective attacks, so we also have to combine counter measures, building on the software development lifecycle described above. The combination of several counter strategies and processes improves overall security and robustness of PoolParty Semantic Suite on all layers, from system level all the way up to user level.

The following sections briefly describe the top 10 security risks identified by OWASP and resulting actions taken in PoolParty to mitigate them.

OWASP Top 10 (2017)

(see also https://www.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf p4 for changes from 2013)

A1:2017-Injection

> Injection flaws occur when untrusted input (e.g. from the web application) is processed as-is, handing over potentially malicious commands hidden in the data unchecked to the system. This can lead to data loss, data corruption, system outage, data exploits (data breach), or even takeover of the whole server.

To prevent injections attacks, input data is never trusted and always validated and pruned (following the principle "never trust the client"). For this, PoolParty uses (XSS) filters, whitelisting, strict validation, consistency checks and other measures on any external data before doing any further processing.

PoolParty Semantic Suite also uses state-of-the-art strategies in custom code and makes use of proven and well established frameworks (Spring Web, Jackson, React) to detect any unsolicited or malicious data.

A2:2017-Broken Authentication

> Attacker uses leaks or flaws in the authentication or session management functions (e.g., exposed accounts, passwords, session IDs) to impersonate users.

PoolParty Semantic Suite's authentication and session management is implemented based on a well-established security framework (Spring Security), offering a single set of control mechanisms that are regularly reviewed and improved (e.g. by updating to the latest stable version and making use of improvements like new and more secure authentication mechanisms).

Optional mechanisms, such as the enforcement of stronger password policies, and integration with external authentication providers (e.g., Single-Sign-On via SAML protocol), offer a broad range of possibilities for adapting individual customer's authentication policy needs.

PoolParty 7 offers additional measures such as password history (enforcing the use of new passwords), or the integration of captchas when an account is locked after a configurable amount of login attempts.

Roadmap:

- Evaluate use of 2-factor-authentication and hardware tokens.
- Keep up-to-date blacklists (e.g., passwords available from data breaches) and block use of bad passwords such as [1]

[1]https://github.com/danielmiessler/SecLists/tree/master/Passwords

A3:2017-Sensitive Data Exposure

> When sensitive data is not protected properly, it might be exposed to malicious parties and lead to data fraud, identity theft, or other crimes. It is therefore paramount to ensure strong encryption and protection of data when transmitted to the browser or other external systems.

PoolParty Semantic Suite follows the industry best practice to not only expose data needed for the particular purpose at hand, but also to only store sensitive data in memory or unencrypted as long as absolutely necessary.

Any data that is eventually exposed to the outside is always encrypted (or hashed, in case of passwords), based on state-of-the-art cryptographic tools and strong standard algorithms, such as Argon2 [1] for hashing passwords or AES with a configurable key size (only limited to the Java version used) for handling third-party credentials.

The use of HTTPS is now strongly encouraged throughout PoolParty to eliminate any man-in-the-middle attacks and ensure a secure communication channel from browser or other clients to PoolParty Semantic Suite server.

Furthermore, PoolParty Semantic Suite allows admins to purge all data, including sensitive user data, like user credentials and history data.

[1] https://www.cryptolux.org/index.php/Argon2

A4:2017-XML External Entities (XXE)

> Unvalidated external XML may be subject to malicious exploits embedded into XML as entities and result in severe security vulnerabilities.

This relatively new threat is mitigated by strict input validation (see also section A1), white listing of allowed XML/HTML constructs, usage of latest framework or library versions and security testing. PoolParty does not process any external XML verbatim, without prior validation or pruning. React UI uses means to prevent XXE attacks, legacy UI code is refactored and isolated.

A5:2017-Broken Access Control

> Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification or destruction of data, or performing a business function outside of the limits of the user.

PoolParty Semantic Suite employs various strategies and safeguards to make sure that access control cannot be circumvented or misused. This includes URL validation, securing end points of the REST API via token, checking of access rights on the server, session control with proper expiration, CORS configuration and more.

On the protocol layer, PoolParty Semantic Suite uses the latest Spring-web framework (5.x) to ensure the underlying implementation for client access is safe and secure.

On the application layer, RDF data is protected by project based access control, providing a single set of restrictions integrated in the architecture of PoolParty Semantic Suite itself. Hence, access control is applied to every request to ensure that the client is authorized to access the requested object.

Roadmap:

- usage of JSON Web Tokens (JWT) to avoid using simple tokens and still allow secure exchange of metadata (e.g. basic user information, roles)
- replace Basic Auth secured REST API with endpoints that require authentication, e.g. by calling an API credentials endpoint that returns a JWT for a given session
- permission-based access controls are going to allow more fine-grained access control mechanisms for resources.

A6:2017-Security Misconfiguration

> Security misconfiguration includes unpatched or outdated components, using default accounts or credentials, enabling unconfigured or unneeded features, and is related to all levels of an application stack, including the platform, web server, application server, database, framework, and custom code.

PoolParty Semantic Suite comes with a dedicated set of licensed and well-defined components. Configuration is taken very seriously and documented extensively. Default parameters are only provided for non-sensitive information to provide a smooth set-up and to help users get on-boarded quickly.

To reduce the amount of configuration issues, development, testing, and system operations teams work together to ensure correct and secure configuration across the entire development and deployment life-cycle - from initial design all the way to production. This includes the following measures and steps:

- Detailed and up-to-date installation and migration
- Documentation of relevant components (including deployment notes, configuration changes, etc.)
- Recommended configuration (c.f., [1]) with restrictive and secure defaults
- Timely hotfix releases and frequent feedback loops between system operations and development to harden deployment configurations
- User-friendly error messages without potentially sensitive information (stack traces and detailed technical information is only provided in the server logs which are only accessible to administrators)

[1]https://help.poolparty.biz/doc/administrator-guide/poolparty-administration/security-configuration/provide-secure-backend-for-ppt-linux

A7:2017-Cross-Site Scripting (XSS)

> XSS flaws occur when an application includes user supplied data in a page sent to the browser without properly validating or escaping that content.

All communication between server and client (UI) enforces output escaping of untrusted HTML data, and any framework or library used is subject to careful selection and maintenance process.

State-of-the-art frameworks such as React, are already secured against XSS or similar attacks by escaping and restricting any HTML output. Also, a strict content security policy is in place, preventing any external code to be executed. This goes hand in hand with injection safe guards mentioned in A01.

To mitigate the attack vectors in respect to XSS attacks, regular awareness trainings are conducted to facilitate secure coding throughout development.

Roadmap:

- Extend the use of tools for automated testing in case of XSS style attacks

A8:2017-Insecure Deserialization

> Applications and APIs will be vulnerable if they deserialize hostile or tampered objects supplied by an attacker.

PoolParty Semantic Suite does not use any low-level serialization over the wire such as RPC, but instead relies on the semantic web standard RDF, a plain text format used in the entire persistence layer. These Strings (RDF triples) are converted using the latest version of RDF4j and a custom, proven and tested mapper. For the document index, Solr/ElasticSearch is used, which is also kept in sync with security patches and constantly migrated to the latest stable version.

The HTTP REST API only exchanges JSON text based objects, which only contain needed data fields that are again filtered and validated (expected type, length, etc.). On the server side, the open source industry standard Jackson mapper ensures strict and consistent processing of any data exchanged.

A9:2017-Using Components with Known Vulnerabilities

To mitigate the risk of attacks by exploiting known security vulnerabilities in publicly available software components, all dependencies (components, libraries, frameworks) are constantly monitored for known security flaws. Hotfixes are also applied in a timely manner. This is ensured by:

- Documenting third-party dependencies and adopting a common version throughout the system
- Monitoring of third-party dependencies in public databases, project mailing lists or security bulletins
- Regularly checking third-party dependencies, using vulnerability scanners

A10:2017-Insufficient Logging & Monitoring

> Attackers rely on the lack of monitoring and timely response to achieve their goals without being detected.

Logging is an integral part of the PoolParty Semantic Suite, as it is also essential for debugging and development.

From a security perspective, it is even more important, because a consistent logging and monitoring layer enables quick and thorough detection of suspicious system activities and enables system operations to act quickly, preventing further damage even when a particular exploit has been successful.

PoolParty 7 provides logging for successful or unsuccessful authentication, API access, essential parts of data being transferred and functionality accessed.

Subsequent versions will build on that basis and provide automatic alerting and distributed logging, which enables monitoring in a clustered environment, following the entire flow for every outside action throughout the system and back.

Roadmap:

- Adoption of a federated logging solution (ELK stack)
- Integration with common monitoring and alerting endpoints

OWASP Summary

Risk	Current mitigation strategies	Roadmap
A1:2017 Injection	• XSS filters, whitelisting, validation and consistency checking • usage of current and state-of-the-art frameworks and libraries such as React, Spring Web and Jackson • external data is never trusted	• moving legacy UI code to React • single service layer for input validation
A2:2017 Broken Authentication	• a single set of strong authentication and session management controls based on well-established security frameworks such as Spring Security • Regular audits of the default security configuration • optional password strategies, enforcement of stronger user passwords, password history and captchas • usage of state-of-the-art hashing (Argon2) and encryption (AES-256) • optional integration with authentication providers, e.g. by using LDAP or SAML protocol	• upgrade to JDK 11 and enforcing 256-bit AES security • support of 2-factor-authentication and hardware keys • keeping up-to-date blacklists and checking user passwords against compromised passwords
A3:2017 Sensitive Data Exposure	• integration of state-of-the-art cryptography including Argon2 hashes and strong dynamic AES encryption • sensitive data can be purged and is kept in the system only where necessary and as short as possible • regular review of cryptography used in PoolParty and refactoring of legacy code • usage of HTTPS for the PoolParty server is fully supported, strongly encouraged, easily configurable and extensively documented	• deprecation of HTTP (following the move in browsers) • migration of weak passwords and hashes to strong up-to-date variants • review error messages for potentially security sensitive details (e.g. Exceptions on the API level, user errors)
A4:2017 XML External Entities (XXE)	• strict input validation and constraining through whitelisting of XML/HTML • usage of latest UI and backend libraries (React, Jackson) • XML data is never processed verbatim	• more automated testing regarding XXE vulnerabilities

A5:2017 Broken Access Control	• state-of-the-art security layer based on Spring Security 5.x • support of role, groups- and project-based access control for every object in PoolParty in a single, uniform service layer • URL validation, secured endpoints, CORS protection	• fine-grained, permission-based access control • usage of JWT instead of simple tokens for better API security
A6:2017 Security Misconfiguration	• cautious and restrictive configuration and deployment • timely hotfix releases for integrating internal and external security patches • security documentation for all PoolParty components and update cycles • security audits and knowledge transfer	• extend automated tests for more coverage of security relevant cases
A7:2017 Cross-Site Scripting (XSS)	• output escaping • awareness training • regular updates/patches of libraries • constant refactoring of legacy code	• extend the use of tools for automated testing for XSS style attacks
A8:2017 Insecure Deserialization	• avoiding the use of low-level, direct (de)serialization and relying on text-based formats (RDF, JSON) that are mapped in a restrictive manner using state-of-the-art frameworks (React, Jackson) • input/output validation, restriction of fields transferred	• phase out legacy frameworks and related code
A9:2017 Using Components with Known Vulnerabilities	• documenting third-party dependencies and adopting a common version throughout the system • monitoring of third-party dependencies in public databases, project mailing lists or security bulletins • regularly checking third-party dependencies by vulnerability scanners	• use Sonatype Nexus plugins and build tools to scan our repositories for known vulnerabilities automatically
A10:2017 Insufficient Logging & Monitoring	• extensive logging of security relevant actions throughout the system, e.g. API access, successful/unsuccessful authentication, data constraint violations, etc.	• adoption of a federated logging solution (ELK stack) • integration with common monitoring and alerting endpoints