SYNERGY® CLOUD

Application Design & Security

Contents

1. Introduction 3
2. Authentication 3
3. Authorization and Access Control 3
4. Session Management & Data and Input Validation 3
5. Cross Site Scripting (XSS) 3-4
6. Cross site request forgery (CSRF) 4
7. Command Injection Flaws/SQL 4
8. Clickjacking 4
9. Buffer Overflows 4
10. Web Application and Server Configuration 4-5

1. Introduction

As a web application, Synergy Cloud can provide an organization with quick access to resources; user-friendly interfaces, and effortless deployment to remote users. With additional access, web applications require security to ensure manageable risks for authorized organizations. The content of this document provides a description of the security challenges faced by Synergy Cloud and the knowledge of the security requirements for the web application. The document is structured as a checklist, for each challenge there are specific checkpoints that delineate the security concern. The checklist provides a basis for securing Synergy Cloud and its database from malicious and unintentional abuse. The goal is to determine those threats beforehand and to create counter measures.

2. Authentication

Synergy Cloud provides a secure method of checking the user's email and password. The data is transmitted to the server securely via HTTPS, the email is checked first. If the email exists, then the hash of user's password is computed and compared to the hash stored in the database. *Note that the hashed password cannot be converted back to plain text.*

3. Authorization and Access Control

Synergy Cloud has a strategy in place to protect back-end and front-end data and systems by using roles, credentials, and sensitivity labels. Synergy Cloud implements a secure Group & Permissions feature which allows only those users with correct access granted to be able to view specific data. The roles are assigned during user creation and can be later updated to increase or decrease the level of visibility of the users.

4. Session Management & Data and Input Validation

Synergy Cloud implements JSON Web Tokens (JWT) for authentication and user sessions. The JWT is sent to the browser using HTTP only cookies and are not accessible by any client-side script. JWTs are assigned a secret key so that prediction attacks cannot work against it. Throttling has been implemented to prevent brute-force attacks. Depending on the URL being accessed, the throttling algorithm only allows a certain number of requests per hour/day/week/month. JWTs are renewed each time the user logs in so fixation attacks will not work. Encrypting sessions is effective against interception; randomly assigned session ids protect against prediction; long key spaces render brute-force attacks less successful and forcing assignment and frequent regeneration of session ids make fixation less problematic. Data is validated using Angular 8's validation algorithm specific to each field and the Django Rest Framework uses serializers to validate the data being sent to the server. Data trimming, cleaning, max-length checks and data validation are based on data type (integers, characters, URLs, etc.).

5. Cross Site Scripting (XSS)

Synergy Cloud implements the strongest defense against attacks of this form via input validation to clean data before it is processed. The server validates all data entering the web application against "known good" criteria to greatly reduce chances of successful attack.

Synergy Cloud uses the following methods to validate data:

- Constrain input – decide what is allowed in the field
- Validate data – type, length, format, and range
- Reject “known bad” input – do not rely only on this as it assumes the programmer knows everything that could possibly be malicious
- Sanitize Input – this can include stripping a null from the end of a user supplied string; escaping output values so they are treated as literals, and HTML or URL encoding to wrap data and treat them as a literal

6. Cross site request forgery (CSRF)

Synergy Cloud uses Django Rest Framework which has built-in protection against most types of CSRF attacks. CSRF protection works by checking for a secret in each POST request. This ensures that a malicious user cannot “replay” a form POST to Synergy Cloud and have another logged in user unwittingly submit that form.

7. Command Injection Flaws/SQL

Synergy Cloud uses Django Rest Framework, Django’s querysets are protected from SQL injection since their queries are constructed using query parameterization. A query’s SQL code is defined separately from the query’s parameters. Since parameters may be user-provided and therefore unsafe, they are escaped by the underlying database driver.

8. Clickjacking

Synergy Cloud uses Django Rest Framework that includes Middleware to protect against clickjacking. Modern browsers honor the X-Frame-Options HTTP header that indicates whether or not a resource is allowed to load within a frame or iframe. If the response contains the header with a value of SAMEORIGIN then the browser will only load the resource in a frame if the request originated from the same site. If the header is set to DENY then the browser will block the resource from loading in a frame no matter which site made the request.

Django provides a few ways to include this header in responses from your site:

- Middleware that sets the header in all responses
- A set of view decorators that can be used to override the Middleware or to only set the header for certain views
- The X-Frame-Options HTTP header will only be set by the Middleware or view decorators if it is not already present in the response

9. Buffer Overflows

Synergy Cloud uses secure data validation protocols to make sure that data being sent to the server is correct and all code that accepts input from users via an HTTP request is reviewed to ensure that it can identify large input. In case the data is too large an error is logged, and data dropped.

10. Web Application and Server Configuration

Synergy Cloud is hosted on Amazon Web Services (AWS) servers, with the highest standards for privacy and data security. All servers a located within the US. Servers are patched regularly with security

updates and only required applications are installed. AWS also provides a strict Access Control List to secure data on servers and databases. Fine-grain identity and access controls combined with continuous monitoring for near real-time security information ensures that the right resources always have the right access, wherever the information is stored. The AWS infrastructure puts strong safeguards in place to help protect data privacy. All data is stored in highly secure AWS data centers.

AWS manages dozens of compliance programs in its infrastructure. The IT infrastructure that AWS provides to its customers is designed and managed in alignment with best security practices and a variety of IT security standards. The following is a partial list of assurance programs with which AWS complies:

- SOC 1/ISAE 3402, SOC 2, SOC 3
- FISMA, DIACAP, and FedRAMP
- PCI DSS Level 1
- ISO 9001, ISO 27001, ISO 27017, ISO 27018