Whitepaper

Stiply Cloud and Development

STIPLY

PART OF 4CEE

Revision history			
Version	**Date**	**Author(s)**	**Description**
1.2	28-12-2023	Jeffrey Bosch	Updated retention period in chapter 8.4
1.1	08-12-2022	Jeffrey Bosch	Made a new version
0.7	07-11-2022	Jeffrey Bosch	Updated architecture drawing and depoment process (5.1.1 & 5.1.2)
0.6	08-03-2022	Jeffrey Bosch	Updated chapter 10.4
0.5	04-02-2022	Jeffrey Bosch	Updated backup scenario, small typography changes.
0.4	05-11-2021	Jeffrey Bosch	Updated secure software development part with new audit steps
0.4	15-09-2021	Jeffrey Bosch	Appleid 4cee template
0.3	17-08-2021	Jeffrey Bosch	The following changes: - Added chapter 5,6 - Updated chapter 7
0.2	10-08-2021	Jeffrey Bosch	The following changes: - Added Chapter 1,2,3,4 - Updated cloud architecture picture
0.1	07-07-2020	Bart Timmersman	Add cloud description

Inhoudsopgave

1 INTRODUCTION

This document describes the technical and operational details of the Stiply Cloud Platform. Both platforms are SaaS services and Stiply uses Amazon Web Services (AWS) as its hosting platform.
The purpose of this document is to give insight into the infrastructure, principles, and security of the Stiply Cloud Platform.

1.1 Cloud team

Stiply is part of 4CEE. To make use of the knowledge and expertise of an ISO-certified cloud team, engineering and maintenance of the Stiply cloud environment are hosted by 4CEE.
If in this document the definition cloud team is used, the cloud team of 4CEE is meant.

1.2 Development team

The development of the Stiply application is managed by the development team of Stiply.

2 Basic Principles

2.1 Continues Delivery

The Stiply software is based on the idea to produce software in cycles, ensuring the software can be released at any specific time. With this idea in mind, we can build, test, and release the software with a greater speed and frequency.

2.2 Scalability

The Stiply software is designed to be scalable, the simple idea is that we have one or multiple workers that are handling the jobs in the background which makes sure that there is no user interference. As we are hosting everything on the AWS Cloud we can scale this to an unlimited amount of resources.

2.3 Secure

The Cloud team is primarily responsible for the security management of virtual networks. The physical networks are maintained by our cloud supplier, Amazon Web Services. We work together to keep the software platform stable and secure. The security measures are described in chapter 9 and 10.

3 Software development and architecture

3.1 Multi-Tenant

Stiply is designed to be multi-tenant by design, this means that we have multiple customers on a single instance of an application running on a single instance of an operating system on a common hardware platform, with only a database.

3.2 Technology Stack

The Stiply software suite has the following stack:

Name	Frontend	Backend
Stiply app	React / JQuery	PHP Laravel
Admin panel	React	PHP Laravel
Carerix	React	Php Laravel Lumen
Word Add	Angular	Stiply app (no backend)
Enterprise portal	Angular	PHP Laravel

3.3 Software Components

The software of Stiply is composed of several software components, all of our sources are maintained in Git repositories. We distinguish the following types:

- Stiply app: our main application that is handling all of our so-called “web” and “API” customers.
- Word Add-in: provide a user interface for Microsoft Word and Office Word 365. The application is served via Amazon Cloudfront to the customers.
- Enterprise Portal: provides a user interface managing Stiply accounts as a customer. The application is served via Amazon Cloudfront.
- Carerix provides an interface for the Carerix software. The application is running on an Amazon EC2 instance.
- Admin our administration app for severing support to our customers. The application is running on an Amazon EC2 instance. Only Stiply employees have credentials to do this system.
- SDK’s: these are our public available SDKs for customers that would like to integrate with our application
  - PHP SDK: for our customer with a PHP codebase
  - .NET SDK: for our customers with a .NET codebase

3.4 INTERACTION BETWEEN SOFTWARE COMPONENTS

The following diagram gives a high-level overview of how the software components work together.

4 Secure software development

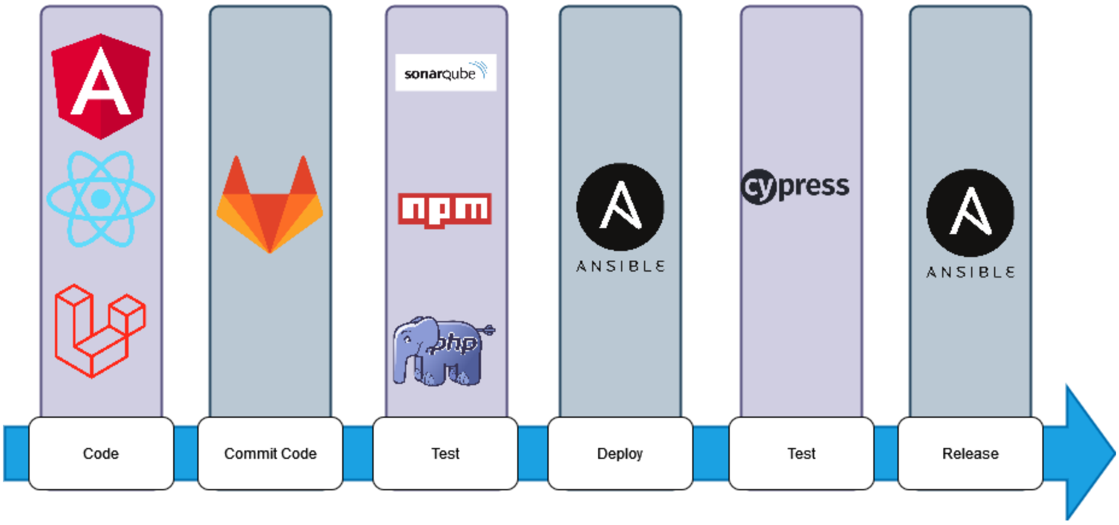
The goal of our build pipeline is to convert code to executable software ready to be deployed to production in a controlled manner. Risks are mitigated by using state-of-the-art components to define quality gates.

Regardless of which part of our software stack is modified, all code is committed to our secured Gitlab environment where a peer review of the code is conducted. Additionally, code quality is monitored with the static analysis engine of SonarQube. Functional correctness is tested with several types of automated tests. Only when all tests pass the code, the code can be deployed to our beta environments. When accepted we will merge all the changes to our main branch from where we can deploy our software to our production environment.

To maintain our vulnerabilities in software we check every week on vulnerable packages. We do this with default audit tools that are available in the package managers. We then report the audit report in our team's channel and add an issue in case we need to update a package because of a vulnerable issue.

The deployment step itself is always a manual approved by a developer, with if needed a cloud engineer standby. When a cloud engineer is on standby is determined by the impact of the code, for example, a database change is a part where a cloud engineer would be on standby.

4.1 Software development process



4.2 Software development tools

Tool	Description
.NET Core	NET Core is a free and open-source, managed computer software framework for Windows, Linux, and macOS operating systems.
Angular	Angular is a TypeScript-based open-source web application framework led by the Angular Team at Google and by a community of individuals and corporations.
Cypress	Cypress is an open-source front-end testing tool.
GitLab	GitLab is a web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking, and CI/CD pipeline features, using an open-source license.
SonarQube	SonarQube is an open-source platform developed for continuous inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells, and security vulnerabilities.
PHP Stan	An open-source tool that's focuses on findings errors based on static analyses
React	React is an open-source library that focuses on building user interfaces, it is maintained by Facebook and the community. It can work with Typescript or JavaScript
NX	NX is a smart and extensible tool that helps you scale your applications or libraries, it supports react, angular, and other typescript languages.
NPM	Node package manager, our node / javascript package manager
PHP Local checker	A vulnerability scanner for our PHP packages.

5 Secure software deployment

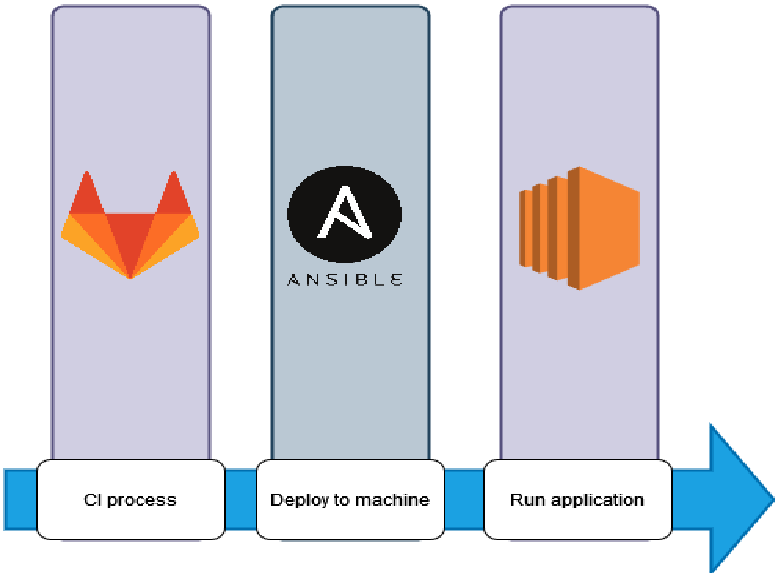
The goal of our deployment pipeline is to deploy stable software to our platform.

Our continuous integration server publishes our software to multiple destinations but all with the same approaches. This allows for a standardized pipeline for all our software, leveraging the strengths of industry-standard components.

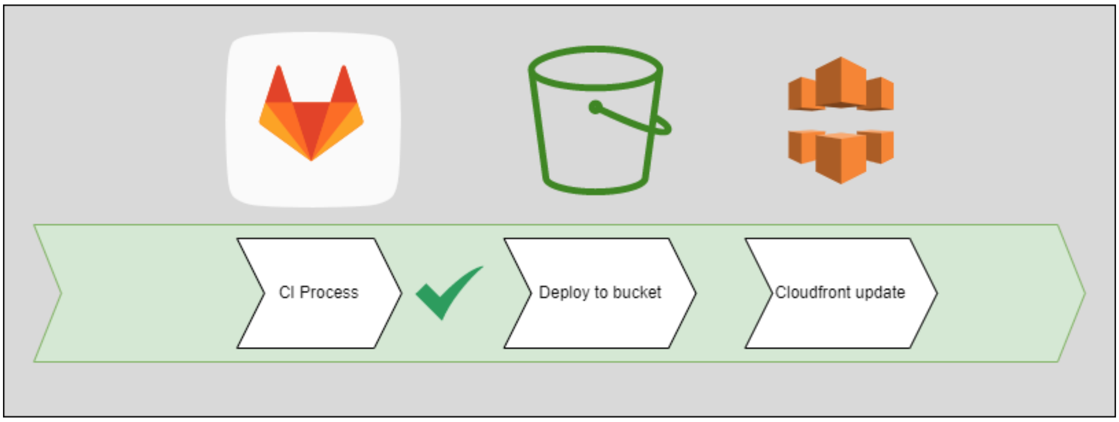
5.1 Software deployment process

The stiply product deployment is two different ways this is intentional as the tech stack is a little bit different.

5.1.1 Laravel applications



5.1.2 Single page apps



5.2 Software deployment process

Tool	Description
GitLab	GitLab is a web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking, and CI/CD pipeline features, using an open-source license.
Ansible	Ansible is an automation platform where we track different automation in so-called playbooks. These playbooks are run on different machines for deploying new software

6 Testing Software

To ensure stable and user-friendly functionality we rigorously test every change. We hereby adhere to industry standards and best practices. We will disclose some of the test types we apply, but before that, we will lay some theoretical groundwork.

6.1 THE THEORY BEHIND OUR VISION ON CORRECT SOFTWARE TESTING:

- Shift left testing vs shift right testing
- Agile testing quadrants

6.1.1 SHIFT LEFT TESTING VS SHIFT RIGHT TESTING

In the early years of the 2000's software testing in general was an afterthought in software development. Its placement right before delivery deadlines brought some understandable drawbacks. The term Shift left testing was coined by Larry Smith in 2001[1] and was used to describe the placement of testing activities as soon as possible in the software lifecycle. Early defect detection results in lower correction cost[2][3] A large enabler for this shift is a high degree of test automation. Developing automated tests are a mandatory part of our development lifecycle.

While shift left testing implies a solid position for testing during development, after deployment, there is still a need to detect issues quickly in production, introducing shift right testing, sometimes also called: 'Testing in production'. By introducing a dev-ops way of working we incorporated the managing and monitoring of the software after delivery. This allows for much faster detection of issues in a production environment, thereby further shortening the feedback loop.

[1] Smith, L. (2001). Shift-Left Testing. Dr. Dobb's Journal, 26, 56–62.

[2] Boehm, B. W. (1981). Software Engineering Economics. USA: Prentice-Hall.

[3] McConnell, S. (2004). Code Complete. USA: Microsoft Press.

6.1.2 AGILE TESTING QUADRANTS

Many testing methodologies adhere to a risk-based approach[4][5][6]. To facilitate comprehensive risk coverage, we use the agile testing quadrants as refined by Janet Gregory and Lisa Crispin in their seminal book on software testing: "More Agile Testing"[7]. They describe two viewpoints for software:

a. Technology facing VS business-facing

b. Guiding development VS critiquing the product

When placed on opposite axes the following matrix with four specific quadrants arises:

- Q1 Technology facing and guiding the development
  The software has a technical quality that needs to be addressed early in development by developers. Testing at this level is used to prove the correctness of small pieces of code (units), integration of units, and adherence to coding standards.

  Q2 Business facing and guiding the development
  Software is designed to serve a specific purpose. The quality of these requirements should be proven as early as possible, even before the coding process starts. This can be done through validation by customers and domain experts.

- Q3 Business facing and critiquing the product
  Once a working piece of software is delivered, its functionality and usefulness can be actively measured. This can be done manually for usability aspects and automated for repetitive tasks.

- Q4 Technology facing and critiquing the product
  There are other quality aspects of the software which do not specifically pertain to documented requirements, like security and performance. These aspects are best suited to test manually with the assistance of specialized tools.

We employ multiple test types per quadrant, thereby combining their respective strong points to a chain of complementary tests.

[4] van der Aalst, L., Davis, C., & van der Aalst, L. (2012). TMap NEXT in scrum: effectief testen in Agile projecten. Netherlands: Kleine Uil, Uitgeverij.

[5] Bouman, E. (2008). SmarTEST, Slim testen van Informatiesystemen. Netherlands: Academic Service.

[6] Koomen, T., van der Aalst, L., Broekman, B., Vroon, M., & van der Aalst, L. (2013). TMap Next: voor resultaatgericht testen. Netherlands: Kleine Uil, Uitgeverij.

[7] Gregory, J., & Crispin, L. (2014b). More Agile Testing (1ste editie). USA: Pearson Education (Us).

6.2 APPLIED TEST TYPES

Having expanded on two viewpoints demonstrating the necessity for multiple complementary test types, we can make a framework for describing our testing strategy.

The following overview references the shift left and right of testing by mentioning the development phase in which it is executed, demonstrating full coverage hereof. Additionally, the coverage of the testing quadrants is made clear by referencing them per test type.

Please note that this overview is meant to give a broad overview of our test strategy, but a complete and detailed description of every test activity.

Test type	Quadrant	Development phase	Description	Test type
Unit test	Q1	Development	We aim to cover as many units of code as possible to automatically detect errors as soon as possible.	Unit test
Prototyping	Q2	Development	One of the first things we create are prototypes of the software to be developed. These are used to elicit feedback from customers regarding functionality and usability. Usually, multiple refinements are done before the actual build starts.	Prototyping
Integration test	Q3	Acceptance	For every release of our software, we execute a fully automated regression test of multiple flows in our software. User actions are simulated in the user interface in a production-like environment.	Integration test

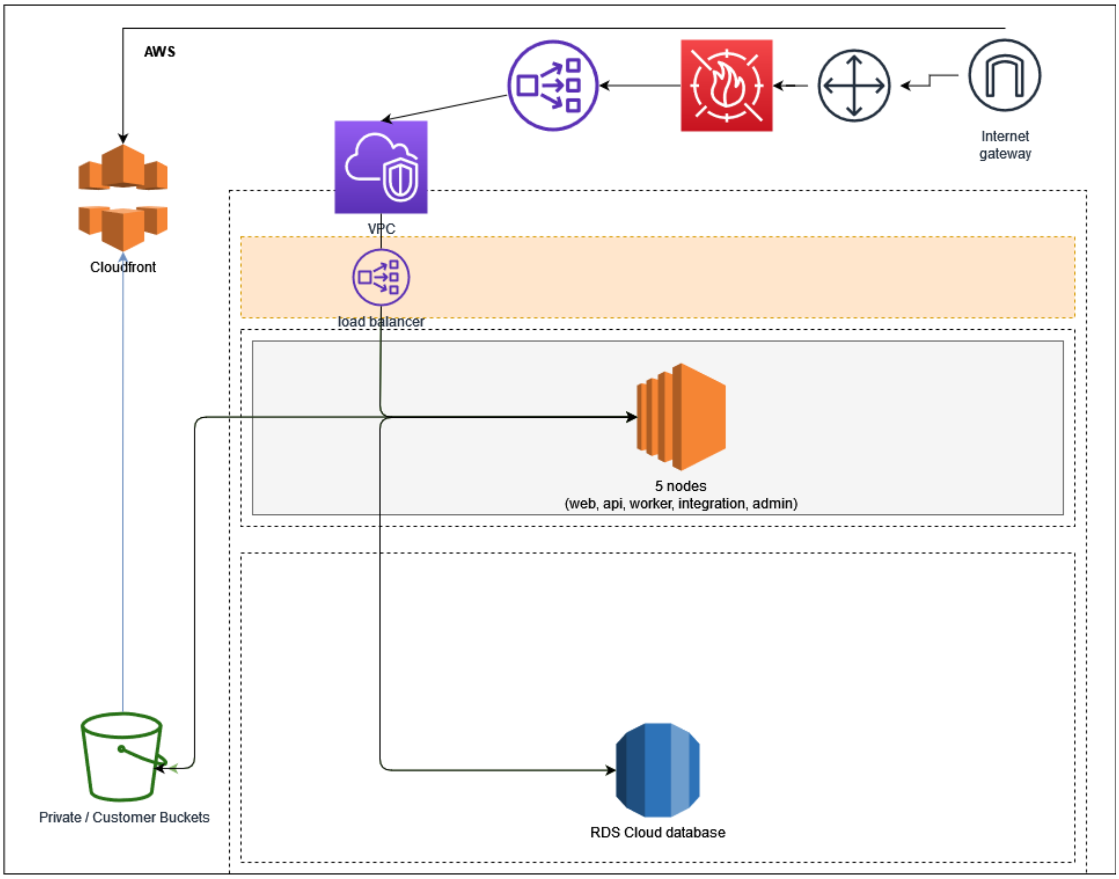
Pentest	Q4	Production	We use checklists to check and improve specific security attributes of our software and infrastructure. Ad hoc independent *penetration tests*, *pentests* for short, are executed.	Pentest
Monitoring	Q4	Production	A multitude of monitoring tools is in use in our production environment. More	Monitoring

6.3 TEST FOLLOW-UP

Having good test coverage is one thing, but without follow up they are pointless. Every automated test is implemented as a quality gate in our build pipeline, meaning when a test fails, the software can't proceed to the Next stage and can't be rolled out to production. Every test has extensive reporting to quickly identify root causes.

7 PLATFORM ARCHITECTURE

The Stiply cloud computing environment (or Stiply Cloud) is running on Amazon Web Services (AWS). Both Stiply and AWS have a shared responsibility, where Stiply has responsibility for security in the cloud, while AWS has responsibility for security of the cloud. This is described in AWS' shared responsibility model (for more information see: https://aws.amazon.com/compliance/shared-responsibility-model)
The Stiply Cloud consists of one AWS organization, the different environments are divided by several AWS account(s) and each has its own VPC.



7.1 COMPONENTS WITHIN THE PLATFORM ARCHITECTURE

The AWS platform contains several components which are used in the infrastructure. The relevant components are described below.

7.1.1 Private/Public subnets

A subnet is an IP address range. It is used for dividing a network into two or smaller networks. It increases routing efficiency and enhances the security of the network. The customer instance is assigned with a private subnet only. Because it needs to be accessible from the internet a Load Balancer is used to route the traffic to the customer instance (through a specific port and encrypted by an SSL certificate).

7.1.2 Internet Gateway

The Internet Gateway is used to route communication from instances to the internet, and vice versa. For the instances, the outbound internet traffic is routed through the Internet Gateway and all inbound internet traffic is blocked. Outgoing traffic is only allowed if the connection is initiated from the server and uses secured protocols (such as SFTP, HTTPS, etc.).

7.1.3 Security groups

Configuration rules for inbound and outbound traffic (allowed communication protocols and ports) are realized within AWS security groups. Each server within the AWS infrastructure is assigned to at least one security group. Security groups can only be changed by AWS admin user accounts.

7.1.4 Regions

The Stiply cloud infrastructure is hosted in Frankfurt, Germany (region EU-central-1). The backups are replicated to Paris, France (region EU-west-3). SMTP services (Mailgun) are hosted in Europe.

7.2 ENVIRONMENTS

The Stiply Cloud infrastructure consists of 2 environments: development and a production environment, these 2 environments are used to ensure a structured roll-out of software updates, upgrades, and/or patches.

Development environment

The development environment is used for developing new functionalities but is also the first environment where new software versions are rolled out. This could either mean new software versions but also new Linux versions/updates or, for example, a new DBMS version. Once we have determined a successful rollout on the development environment, the software will be rolled out to the test environment. The development environment is not available for the customer and is only meant for internal use.

8 Disaster recovery

For disaster recovery, there are certain procedures in place in the event of a human error, data corruption, or failing server instance. Stiply’s strategy for Disaster Recovery is implemented with the use of a backup process (see paragraph “backup model” for a description). This process allows the Easy Systems Cloud team to recover system operation, with the defined RPO (see paragraph 6.1.2) and RTO (see paragraph 6.1.2).

8.1 Procedure

In case of a disruption/disaster, a procedure is in place to determine the impact and to resolve the disruption within the agreed SLA. In case of service disruption within business hours (08:30 – 17:00 CET/CEST), the Cloud team will be notified directly. Notifications will be sent out automatically throughout multiple channels.

Escalation levels are in place:

1. The stiply support engineer
2. The cloud team
3. The head of development and operations
4. Customers, if it will affect their business

The following events will take place in case of a disruption/disaster:

8.1.1 Data loss analysis

An investigation will be performed to determine if there is any data loss due to the disruption. In case of data loss, an overview will be created of the data that is lost and has to be recovered.

8.1.2 Data recovery plan

After the analysis of the data loss, a plan is made on how to recover the data. This could be by reprocessing data or restoring a backup.

8.1.3 Expected recovery time

After analysis of data loss and verification of the recovery plan, an estimated recovery time can be set (also according to the agreed SLA). The recovery time will be communicated to the customer in case the business was affected by the disruption.

8.1.4 Escalation

The outcome of the impact analysis will determine if the disruption will be escalated to the Cloud team lead and/or the delivery manager. Escalation can be used for allocating more resources or to determine the communication plan to the customer.

8.1.5 Communication with customer

In case the business process is affected by the disruption, the customer will be notified by phone and/or email or via our status portal/ website. The Stiply support engineer will be involved to take care of issue management and inform our customer contact.

8.2 RECOVERY POINT OBJECTIVE (RPO)

In case of a major malfunction/disruption, the RPO is set to 12 hours. That means that backups of the database and snapshots from instances will be restored with data that is not older than 12 hours.

In case of smaller disruptions and cases, data can be recovered/restored without the need to use a full backup, the data will be restored manually. Any changes done will be reverted to the point in time the service was working correctly.

8.3 RECOVERY TIME OBJECTIVE (RTO)

In the event of an unrecoverable loss of a data storage device, full backups will be restored of the database and/or server instances. After the restore, checks are made to determine if the restore was successful. A business impact analysis will be made and the outcome will be communicated to the customer. All necessary steps to restore the data and the expected RTO will be included in the communication. The RTO goal is set to 48 hours. Updates on the expected recovery time will be sent to the customer every 4 hours.

Below an overview is provided of the RPO/RTO timeline in case of disaster recovery.

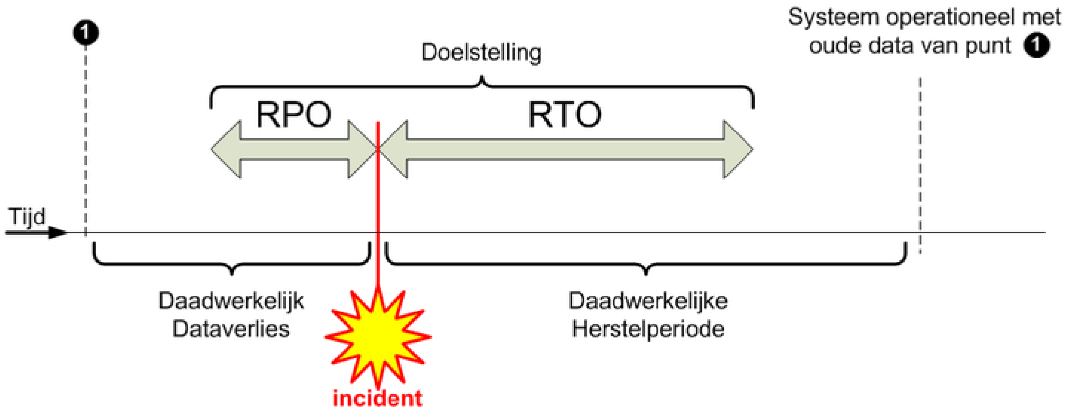


Image 9.1.3 – RPO & RTO

The Overall Recovery goal is the combination of the RPO and RTO. In this time frame, the backup recovery and the data restore are performed. Both the infrastructure and the business processes are fully restored.

Program code is developed at Easy Systems and available outside the SaaS environment. This program code is backed up daily to a remote site.

8.4 BACKUP MODEL

In the Stiplyenvironment, database backups and server snapshots are automatically made every 12 hours. The retention period for the backups is 1 month. That means that backups that are older than 1 month will be automatically removed.
All backup files and snapshots are encrypted with AES-256 encryption. The backup files and snapshots can only be accessed with an encryption key that is stored in the AWS administrator account.

8.4.1 BACKUP RETENTION

The retention period for backups is set to 35 days. After every new successful backup, the oldest backup will be removed (taking the retention period into account).

8.5 FILE BACKUPS

Blobs (images, documents) are stored in the customer instance Amazon S3 bucket. The policy is to keep a copy of the data in the AWS backup region in Paris, France. The bucket has an active replication configuration which replicates the object async to the backup bucket. The backup bucket is owned by a different AWS account. For security reasons, the bucket owner will become the owner of the data after the replication was successful.

9 CLOUD SERVICES

9.1 ORGANIZATION

Cloud Operations is the responsibility of a dedicated Cloud team. The Cloud team is responsible for setting up, maintaining, monitoring, and updating the SaaS platform and its applications. The team is working closely with the Stiply customer support, and development team(s) to ensure the reliability and professionalism of the hosted platform.
Access to the cloud environment is only allowed to persons that are qualified and assigned to Cloud tasks. Access is only provided with personal accounts, all actions/changes in the cloud environment are being logged for traceability and auditability.

9.2 MONITORING

Monitoring is applied at several layers of the Cloud environment. In the following paragraphs is shown on which layers it is applied and how this is done.

9.2.1 Application/software monitoring

The application is monitored on different aspects, to ensure business continuity, performance and to be able to troubleshoot issues. These different aspects are described below:

- Application service

  Applications are monitored on running state. If a service is not running it will be flagged by our monitoring software, according to measures that will be taken.
- The healthiness of the application connection to the database, Redis and

9.2.2 Hardware/server monitoring

For each server within the Stiply SaaS environment, there is hardware/server monitoring in place to ensure availability and acceptable performance. The different counters are described below:

- **CPU**

  The CPU load of the server instance is monitored. An automatic alarm will be pushed by our monitoring software if thresholds are exceeded for a configured period.
- **Memory**

  The physical memory usage of the server instance is monitored. An automatic alarm will be pushed by our monitoring software if thresholds are exceeded for a configured period.
- **Disk space**

  The disk space is monitored for all server disk drives. If the available disk space drops below the threshold (low disk space), an automatic alarm will be pushed by our monitoring software
- **Network**

  All incoming and outgoing network traffic is monitored per server. An automatic alarm will be pushed by our monitoring software if thresholds are exceeded for a configured period.

9.2.3 AWS Web Application Firewall (WAF)

AWS WAF is a web application firewall that helps protect your web applications or APIs against common web exploits that may affect availability, compromise security, or consume excessive resources. AWS WAF gives you control over how traffic reaches your applications by enabling you to create security rules that block common attack patterns, such as SQL injection or cross-site scripting, and rules that filter out specific traffic patterns you define. You can get started quickly using Managed Rules for AWS WAF, a pre-configured set of rules managed by AWS or AWS Marketplace Sellers. The Managed Rules for WAF address issues like the OWASP Top 10 security risks. These rules are regularly updated as new issues emerge. AWS WAF includes a full-featured API that you can use to automate the creation, deployment, and maintenance of security rules.

With AWS WAF, you pay only for what you use. The pricing is based on how many rules you deploy and how many webs requests your application receives. There are no upfront commitments.

You can deploy AWS WAF on Amazon CloudFront as part of your CDN solution, the Application Load Balancer that fronts your web servers or origin servers running on EC2, or Amazon API Gateway for your APIs.

Source: https://aws.amazon.com/waf/

9.2.4 AWS Security Hub

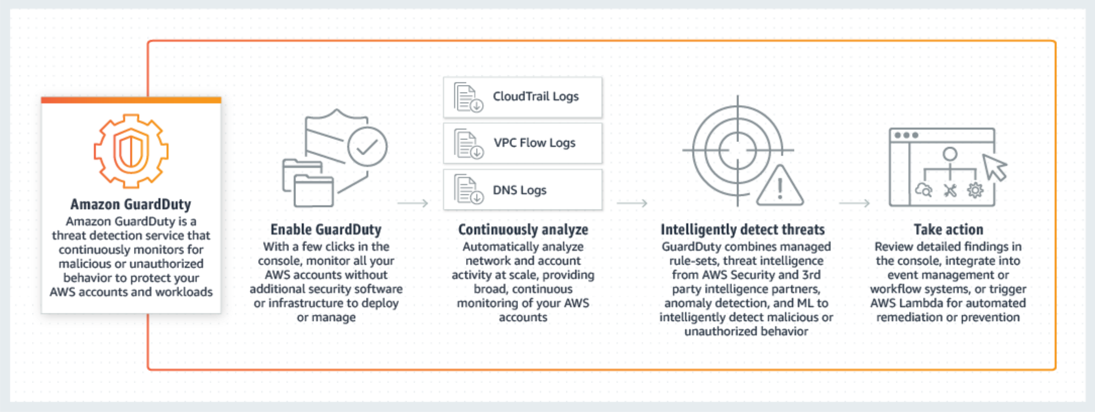
AWS Security Hub gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts. There is a range of powerful security tools at your disposal, from firewalls and endpoint protection to vulnerability and compliance scanners. But oftentimes this leaves your team switching back-and-forth between these tools to deal with hundreds, and sometimes thousands, of security alerts every day. With Security Hub, you now have a single place that aggregates, organizes, and prioritizes your security alerts, or findings, from multiple AWS services, such as Amazon GuardDuty, Amazon Inspector, Amazon Macie, AWS Identity and Access Management (IAM) Access Analyzer, and AWS Firewall Manager, as well as from AWS Partner solutions. AWS Security Hub continuously monitors your environment using automated compliance checks based on the AWS best practices and industry standards your organization follows. You can also take action on these security and compliance findings by investigating them in Amazon Detective or by using Amazon CloudWatch Event rules to send the findings to ticketing, chat, Security Information and Event Management (SIEM), Security Orchestration Automation and Response (SOAR), and incident management tools or to custom remediation playbooks. Get started with AWS Security Hub in just a few clicks in the Management Console and once enabled, Security Hub will begin aggregating and prioritizing findings and conducting compliance checks.
Source: https://aws.amazon.com/security-hub/

9.2.5 Amazon Inspector

Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. Amazon Inspector automatically assesses applications for exposure, vulnerabilities, and deviations from best practices. After performing an assessment, Amazon Inspector produces a detailed list of security findings prioritized by level of severity. These findings can be reviewed directly or as part of detailed assessment reports which are available via the Amazon Inspector console or API.
Amazon Inspector security assessments help you check for unintended network accessibility of your Amazon EC2 instances and vulnerabilities on those EC2 instances. Amazon Inspector assessments are offered to you as pre-defined rules packages mapped to common security best practices and vulnerability definitions. Examples of built-in rules include checking for access to your EC2 instances from the internet, remote root login being enabled, or vulnerable software versions installed. These rules are regularly updated by AWS security researchers.
Source: https://aws.amazon.com/inspector/

9.2.6 Amazon GuardDuty

Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behavior to protect your AWS accounts and workloads. With the cloud, the collection and aggregation of account and network activities are simplified, but it can be time-consuming for security teams to continuously analyze event log data for potential threats. With GuardDuty, you now have an intelligent and cost-effective option for continuous threat detection in the AWS Cloud. The service uses machine learning, anomaly detection, and integrated threat intelligence to identify and prioritize potential threats. GuardDuty analyzes tens of billions of events across multiple AWS data sources, such as AWS CloudTrail, Amazon VPC Flow Logs, and DNS logs. With a few clicks in the AWS Management Console, GuardDuty can be enabled with no software or hardware to deploy or maintain. By integrating with AWS CloudWatch Events, GuardDuty alerts are actionable, easy to aggregate across multiple accounts, and straightforward to push into existing event management and workflow systems.



Amazon GuardDuty
Amazon GuardDuty is a threat detection service that continuously monitors for malicious or unauthorized behavior to protect your AWS accounts and workloads

Enable GuardDuty
With a few clicks in the console, monitor all your AWS accounts without additional security software or infrastructure to deploy or manage

Continuously analyze
Automatically analyze network and account activity at scale, providing broad, continuous monitoring of your AWS accounts

Intelligently detect threats
GuardDuty combines managed rule-sets, threat intelligence from AWS Security and 3rd party intelligence partners, anomaly detection, and ML to intelligently detect malicious or unauthorized behavior

Take action
Review detailed findings in the console, integrate into event management or workflow systems, or trigger AWS Lambda for automated remediation or prevention

Image 8.2.2.1 – GuardDuty, how it works

Source: https://aws.amazon.com/guardduty/

9.2.7 AWS IAM Access Analyzer

AWS IAM Access Analyzer helps you identify the resources in your account, such as Amazon S3 buckets or IAM roles, that are shared with an external entity. This lets you identify unintended access to your resources and data, which is a security risk. Access Analyzer identifies resources that are shared with external principals by using logic-based reasoning to analyze the resource-based policies in your AWS environment. For each instance of a resource that is shared outside of your account, Access Analyzer generates a finding. Findings include information about the access and the external principle that it is granted to. You can review findings to determine whether the access is intended and safe, or the access is unintended and a security risk.
Source: https://docs.aws.amazon.com/IAM/latest/UserGuide/what-is-access-analyzer.html

9.2.8 Monitoring tools

Several tools are used to monitor the Stiply cloud environment. Beneath these tools are described.

9.2.9 AWS Security Hub

AWS Security Hub gives a view of your high-priority security alerts and compliance status across AWS accounts. Within the Hub, there are firewalls and endpoint protection to vulnerability and compliance scanners.

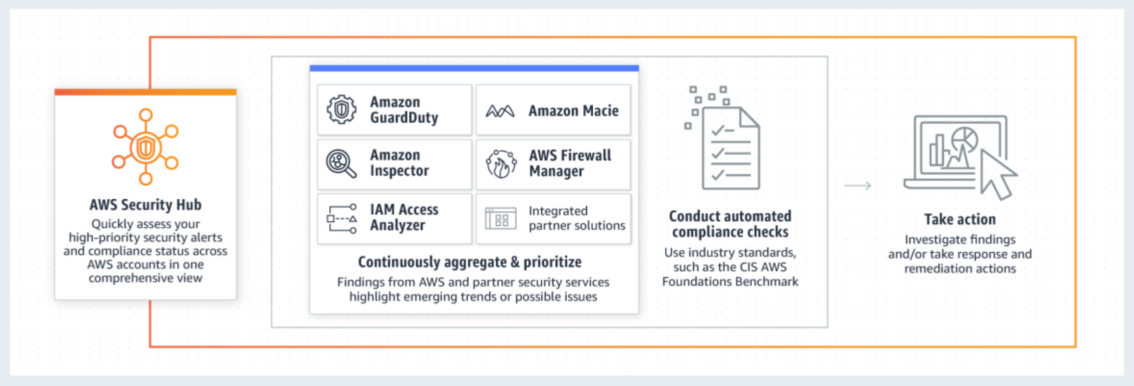

AWS Security Hub
Quickly assess your high-priority security alerts and compliance status across AWS accounts in one comprehensive view
Amazon GuardDuty
Amazon Macie
Amazon Inspector
AWS Firewall Manager
IAM Access Analyzer
Integrated partner solutions
Continuously aggregate & prioritize
Findings from AWS and partner security services highlight emerging trends or possible issues
Conduct automated compliance checks
Use industry standards, such as the CIS AWS Foundations Benchmark
Take action
Investigate findings and/or take response and remediation actions

Image 8.2.4.1.1 – Security Hub, how it works

Servers and networks are constantly monitored for security issues. Alarms will be triggered through AWS GuardDuty and AWS Inspector which associates and analyses events. These alarms will be directly forwarded to the Cloud team so they can act swiftly and accordingly. Alarms are sent to the Cloud team through different notification channels, such as email and Microsoft Teams. Outside business hours, the alarms are also sent to mobile devices.

9.3 MAINTENANCE

Maintenance is performed on application, database, and operating system levels.

9.3.1 Database maintenance

Database maintenance is performed every week, outside working hours. The database maintenance window contains the following steps:

- Reorganize and rebuild indexes: correct problems regarding non-optimal indexing.
- Minor software upgrades
- Slow query analysis

9.4 AVAILABILITY

9.4.1 Cloud/SaaS Availability

Availability of the service means that the referenced environment (production or test environment) is operational and accessible. In particular, that means the correct functioning of all hardware, software, and connectivity components. There is a service level agreement in place for the Stiply SaaS environment. Please refer to the Stiply SLA document for more information

10 SECURITY

Security of the Stiply SaaS service and infrastructure is ensured through the following principles.

10.1 VULNERABILITY MANAGEMENT

The environment is protected against viruses, spam, malicious software, and dangerous active content with the use of AWS WAF - Web Application Firewall, AWS GuardDuty, AWS Inspector, and antivirus software. All emails, data transfers, downloads, uploads, etc. are monitored. An automatic notification will be sent out to the Cloud team in case of a vulnerability. In case of infected files, the customer will be notified as well.
AWS GuardDuty is continuously monitoring the network activity and account behavior in the AWS infrastructure. All events are logged and analyzed and notifications will be forwarded to the Cloud team in case of potential risk. The team will analyze the AWS flow logs and AWS CloudTrail logging and act accordingly if needed.
For security assessments on the server level, AWS Inspector is used. This service is used to detect vulnerabilities and deviations. The “CIS Security Benchmark” is used in AWS inspector as best practice. AWS Inspector will automatically perform an assessment and forward events to the DevOps for further handling.

10.2 ACCESS CONTROL

Access to the Stiply SaaS environment and access to customer data is granted using the least-privilege principle.
Only Cloud team members and the members of the development team have access to the AWS platform environment.

10.2.1 AWS Accounts

Only Cloud team members are given administrator access to the AWS infrastructure/environment. Access is revoked if an employee no longer needs to access the environment. Two-factor authentication is applicable when logging on to the AWS environment.

10.2.2 AWS IAM Roles

Within AWS, user roles are defined. These are called IAM roles. The use of IAM roles allows segregation of duties for users that can access the AWS environment.

More privileged accounts must first be approved by the Stiply management team at least one person before assigning them.

10.3 Contract termination

When a customer terminates the contract it is important to deactivate the customer. This can be done two ways:

1. Customer unsubscribed to the contract and based on the contract we automatically deactivate their account
2. Customers reach out to our support and we do it manually.

The data itself will be cleaned after a period based on some internal cronjob processes.

10.4 DATA PROTECTION (CRYPTOGRAPHIC, ENCRYPTION, HASHING)

10.4.1 Data transfer requirements

After the *Schrems II* ruling of the EU Court of Justice on 16 July 2020, it became clear that data transfers to the United States should be taken into careful consideration. The Court considered that operators that seek to transfer data to the U.S., must ensure that there are effective supplementary measures in place to compensate for lacunae in the protection of the American legal system.

Stiply does not transfer data – such as customer data or data contained in documents – to the United States. However, Stiply's cloud infrastructure is hosted by AWS in the EU. AWS is a service of Amazon EU Sarl, located in Luxembourg. Its parent company Amazon.com, Inc. is based in the U.S. Thus, due to American legislation, Amazon can be compelled to hand over data from its customers (such as Stiply including Stiply's customers) to U.S. governmental bodies. Stiply considered this scenario and took the following measures:

- the data are processed using strong encryption;
- the encryption algorithm and its parameterization are state-of-the-art;
- the strength of the encryption and key length takes into account the specific time period during which the confidentiality of the encrypted personal data must be preserved;
- the encryption algorithm is implemented correctly;
- the encryption keys are reliably managed using AWS Key Management Service (KMS) to manage, generate and rotate cryptographic keys;
- the keys are controlled by Stiply.

In conformity with best practices, Stiply rotates its keys dynamically and the keys are kept separate from the encrypted data. Next to this, there is a strict succession of historical keys to reduce the impact of unlawful data access. All data are kept within the secured Stiply SaaS environment. Data at rest are encrypted with AES-256 encryption. Next to this, we have Privileged Access Management (PAM) in place. Access to the encryption key management service is very limited. Only a small number of employees have access. Only relevant employees will be granted access to the encryption key alongside the AWS IAM role that is assigned. Stiply also uses AWS CloudTrail. CloudTrail monitors and records account activity across the AWS infrastructure. Finally, customer data are segregated on database and application level.

We believe that Amazon's AWS services provide the best data protection for our customers.

For example, as a principal rule, Amazon does not disclose data. And if a governmental body sends Amazon a demand for data, it will redirect that body to the customer (Stiply). Furthermore, AWS cloud services adhere to CISPE Data Protection Code of Conduct. This Code of Conduct has been validated by the European Data Protection Board (EDPB) and approved by the French Data Protection Authority (CNIL) and assures that the AWS services meet the applicable GDPR requirements. Lastly, in its white paper Amazon convincingly substantiates that AWS services comply with the data transfer requirements of the *Schrems II* ruling.

10.4.2 Data in transit

Data in transit, initiated from the Stiply SaaS environment to the internet is encrypted using Transport Layer Security (TLS). We use TLS 1.2 which is a cryptographic protocol designed to provide communications security over a computer network. TLS 1.2 can secure all communications between servers and web browsers. The TLS protocol aims primarily to provide privacy and data integrity between two or more communicating computer applications.

In addition to the cloud infrastructure did we also taken security measures on our applications for example:

1. All our applications are only accessible through the https protocol this means that data up to our API gateway is encrypted by this standard protocol.
2. The passwords used within the applications are hashed. This means that they cannot be traced, not even by us.

10.4.3 Exceptions tailored environment

Our infrastructure is designed and construct to provide the best possible protection. To keep all data secure. We don't make exceptions to our infrastructure when it comes to security measures. There will be no tailormade solution for our customers.

10.5 DATA CLASSIFICATION (CIP TASKFORCE BASELINE)

The Center for Information Security and Privacy Protection (Centrum Informatiebeveiliging en Privacybescherming) published a baseline for data classification. In this baseline, they include a classification guideline. Documents and data about Expenses, Invoices, and Contracts were judged on business and personal impact. The sensitivity is measured by integrity and confidentiality rating, conform the classification guideline.

Information types	Business process impact	Integrity classification	Confidentiality classification
Documents	4	2 – High	2 - Confidential

We only deliver the highest possible setting on security to our documents and data

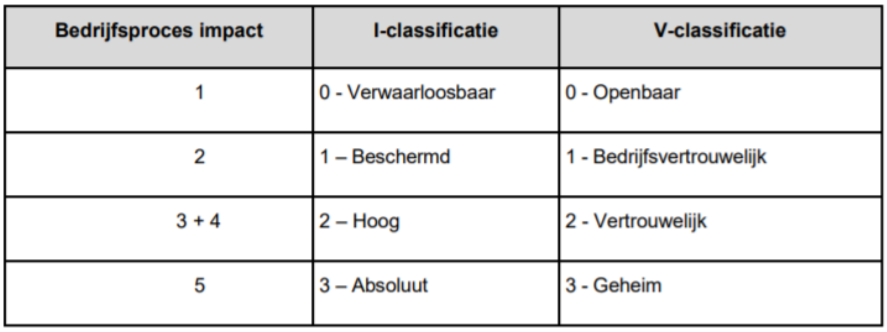
Because the data and documents that are used in our software all scored 4 on business process impact and the Integrity classification is a minimum of 2 – high. The Confidentiality classification is 2 – Confidentiality we decided to set the highest possible setting on security to our documents. Only the people that are authorized in a specific process can see the documents and only the people that are authorized can adjust. This setting is set on processes the documents are set to confidential.

Background information about CIP

CIP was founded by the Dutch tax service, DUO, SVB and UWV and originates from the program Compacte Rijksdienst (2011-2012).[8]

[8] https://www.cip-overheid.nl/media/1166/bid-operationele-producten-bir-010-dataclassificatie-1_1.pdf

Voor de classificatie naar de inzichten **integriteit** en **vertrouwelijkheid** is de vertaling als volgt:



Bedrijfsproces impact	I-classificatie	V-classificatie
1	0 - Verwaarloosbaar	0 - Openbaar
2	1 – Beschermd	1 - Bedrijfsvertrouwelijk
3 + 4	2 – Hoog	2 - Vertrouwelijk
5	3 – Absoluut	3 - Geheim

10.6 INCIDENT MANAGEMENT

Security and software issues and incidents are identified via multiple channels, from technical and functional monitoring to the Stiply customer service desk. Security incident handling follows a defined procedure and is the responsibility of the Stiply security officer. For more details about the Stiply customer service desk incident response times, security incidents, and issue management, please refer to the Stiply SLA.

10.7 RISK AND COMPLIANCE

Easy Systems is ISO 9001, ISO27001, and ISO 27017 certified. Both Stiply and Easy Systems are following all relevant data protection guidelines and rules, applicable to our operations. Stiply has planned to get ISO certified for all three of the certificates in Q2 2022.

10.8 DATA SEGREGATION

Customer data is isolated in separate ways per platform. Beneath the oversight per platform is shown.

11 CLOUD RESTRICTIONS

To ensure a stable working cloud environment there are certain guidelines in place. These guidelines are based on best practices.

11.1 DATABASE

Direct database access or database links between the database and other systems outside the Stiply SaaS environment is not allowed.

11.2 INTERNET GATEWAY

An internet gateway is a network "node" that connects two different networks that use different protocols (rules) for communicating. In the most basic term, an internet gateway is where data stops on its way to, or from other networks.

11.3 SUBNETS

A subnet is an IP address range. It is bound to a single AWS zone and cannot span multiple AWS zones or regions.

11.4 VPC

On-demand configurable pool of shared computing resources, allocated within a public cloud environment.

11.5 IAM

AWS Identity and Access Management (IAM) enables you to securely manage access to AWS services and resources. Using IAM, it is possible to create and manage AWS users and groups and use permissions to allow and deny their access to AWS resources.

11.6 AWS GUARDDUTY

Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behavior to protect AWS accounts and workloads.

11.7 AWS SYSTEM MANAGER

AWS Systems Manager provides visibility and control of the AWS infrastructure. Systems Manager provides a unified user interface that makes it possible to view operational data from multiple AWS services and allows to automate operational tasks across different AWS resources.

11.8 AWS INSPECTOR

Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. Amazon Inspector automatically assesses applications for exposure, vulnerabilities, and deviations from best practices.

12 AUDITS INITIATED BY CUSTOMERS

12.1 ADDITIONAL INDEPENDENT AUDIT

Customers have the right to request an additional independent audit. This audit can be carried out on the technical and organizational security measures that Stiply takes to protect the information of it and its customers. The following conditions are attached to this audit:

- The audit must be planned and approved by Stiply;
- All risks involving this audit must be evaluated and mitigated;
- Stiply will inform involved parties like AWS;
- The audit takes place against the ISO9001 & ISO27001 & ISO27017 standard;
- The audit is performed by an accredited auditor;
- The costs for the audit are for the client;
- All confidential business information remains within Stiply;
- All information about other customers remains within Stiply.
- The additional independent audit can only be requested once a year

Stiply accepts to resolve all identified shortcomings within a reasonable period.

13 TIME SYNC SERVICE

13.1 Redundant Network Time Protocol Sync Service

The cloud infrastructure that we use offers a Time Sync Service “Amazon Time Sync Service”

The Amazon Time Sync Service is a time synchronization service delivered over Network Time Protocol (NTP) which uses a fleet of redundant satellite-connected and atomic clocks in each region to deliver a highly accurate reference clock. The Amazon Time Sync Service automatically smooths any leap seconds that are added to UTC. This service is available in all public AWS regions to all instances running in a VPC.

The service is a consistent and accurate time reference is crucial for many server tasks and processes. Most system logs include a timestamp that we use to determine for example when problems occur and in what order the events take place.

The Amazon Time Sync Service is available through NTP at the 169.254.169.123 IP address for any instance running in a VPC. All our applications and underlying services and components use the time sync service.