Company Name: Keboola s.r.o.

Business Model: Keboola operates a cloud-based ETL tool based on monthly subscription fees.

Tool Access/Use: The tool Keboola Connection is used by clients employees, most commonly data analysts (or similar job positions) to create BI solution, to do data analytics, to prepare logical data model and usually load the data into visualization tool (for ex. GoodData, Tableau, Qliksense, etc.)

Tool Design: https://www.keboola.com/whitepaper
For the reporting aspect, Client will pull metrics from the API into our tool Keboola Connection.

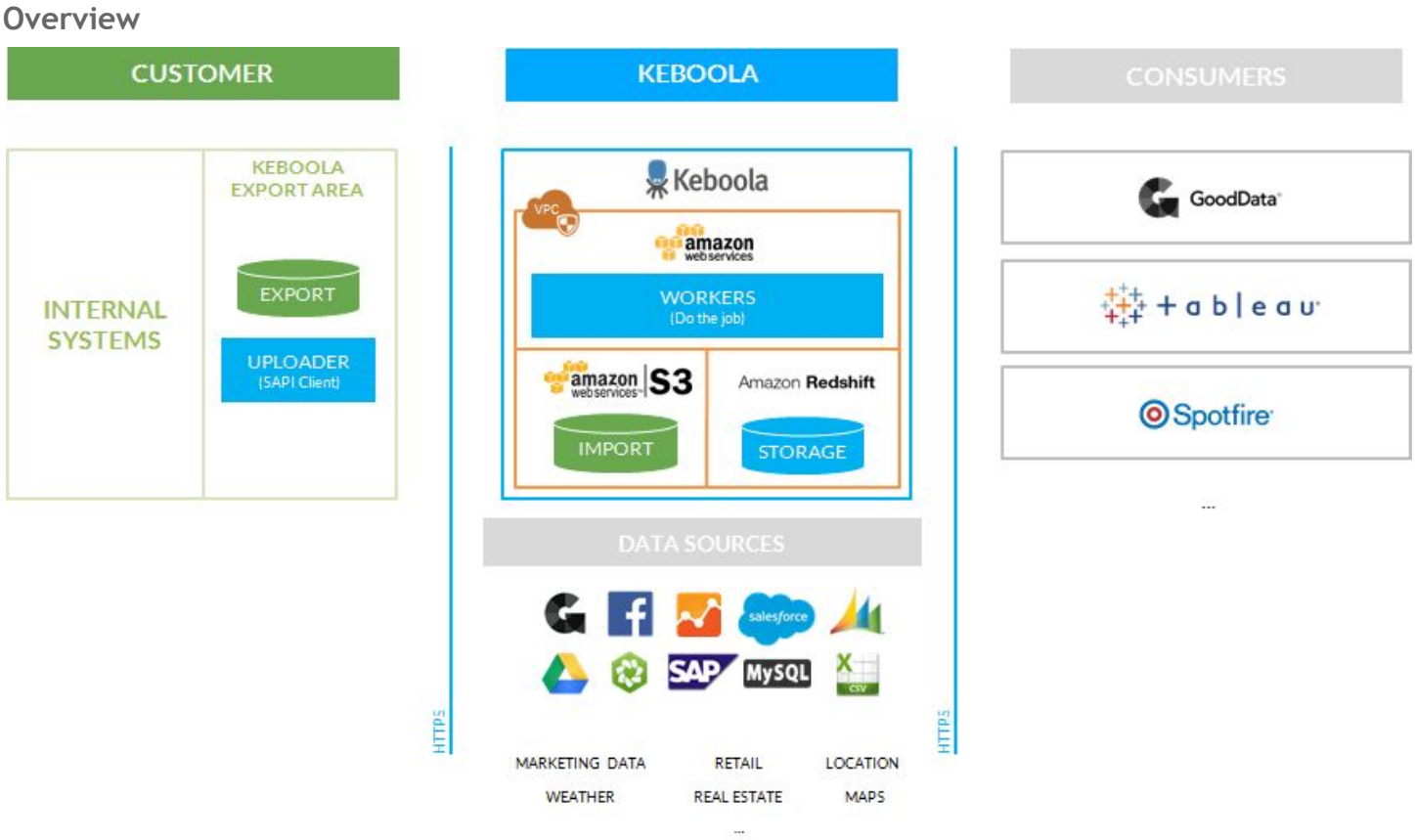



Figure 1: High level overview of Keboola environment and data flows

- Customer
  Denotes customer's infrastructure on customer's premise. In this infrastructure there should be a specific data storage where data extracts are prepared for upload to Keboola.

- Uploader
  is a code provided by Keboola that runs on customer's infrastructure and manages authentication, connectivity and upload of data from extracts to Keboola. This piece of software is open source to be scrutinized, built and deployed by customers.

- Keboola (Keboola Connection)
  denotes all software components of the Keboola Connection solution

- Workers

are components processing your data, let it be transformations, enrichment or analytics.

## Keboola Overview

Keboola is fundamentally built and operated on top of Amazon AWS and other 3rd party services, such as PaperTrail (logging), PagerDuty (alerting) or NewRelic (monitoring). Keboola inherits from Amazon important security characteristics for data storage, encryption, access control, archiving and others. Detailed description of underlying Amazon security concerns are well documented in Amazon whitepapers: https://aws.amazon.com/security/.
Data stored in Amazon (Amazon Redshift) can be encrypted using HSM (hardware encryption, https://aws.amazon.com/cloudhsm/ ). Data storage then complies with PCI DSS, SOX and HIPAA (https://docs.aws.amazon.com/redshift/latest/mgmt/working-with-db-encryption.html ).
Access control within Keboola is based on Amazon IAM and uses two-phase authentication to Keboola. All communication goes through SSL connections calling REST APIs. Connection can be established only with a valid token encrypted with blowfish algorithm. Token authorization is performed by Storage API isolated from the rest of the infrastructure.

## Keboola Architecture

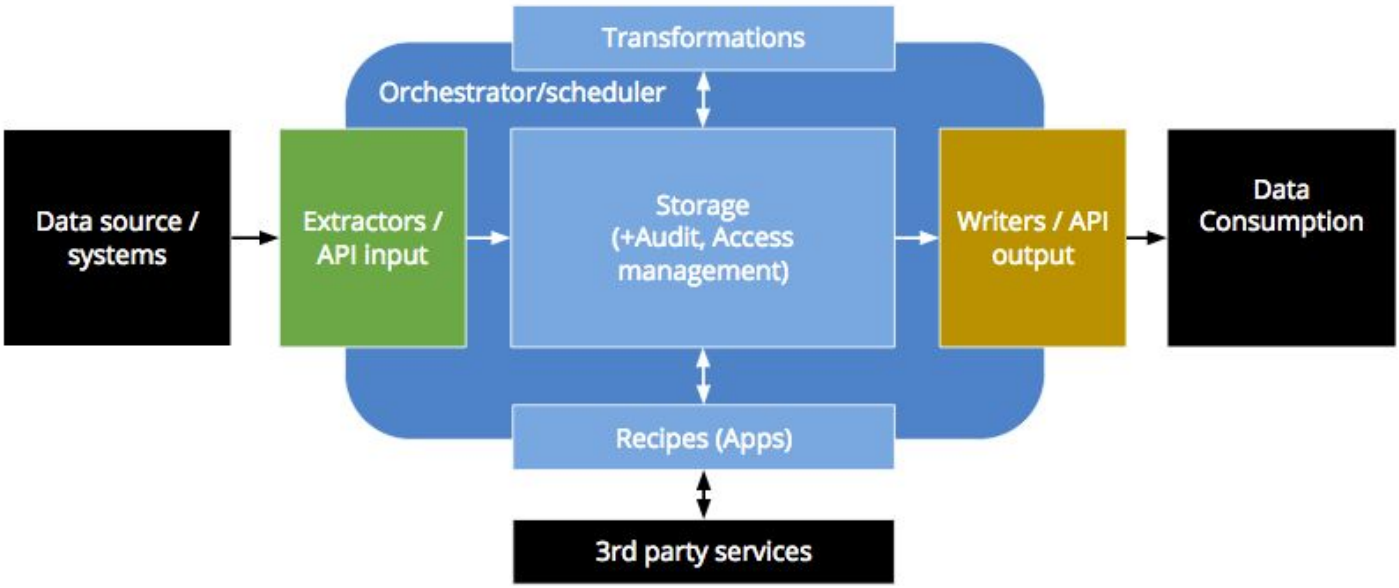



Figure 2: Keboola architecture components

## Security Architecture

Security is the key concern underpinning all fundamental concepts of Keboola Architecture. There is a strict separation and audit mechanism in how data are uploaded and consequently processed and stored.
Data can be uploaded to Keboola in two ways:
By the Uploader - Data prepared by the customer are stored in an export area (with security requirements specified and employed by the customer). Uploader is a customer side component responsible for retrieving data from customer premises and uploading data securely via HTTPS through REST API to Keboola storage in Amazon S3. All data are at first imported to Amazon S3 service where each and every customer has access to a dedicated encrypted S3 bucket. Customer is in full control of the upload functionality. What data, when and where they should be uploaded is configured and initiated by the customer. Uploader is available for all most common platforms (C#, PHP, Java, Javascript, Ruby, Python) either as an executable or as a source code to be scrutinized, built and deployed by the customer.

By an Extractor - A server side component running in Keboola, remotely extracting data from various sources and uploading them via the very same API as the Uploader does. Extractors can be easily configured directly in Keboola UI.
Every data upload can be performed with a valid token only. For more see Authentication & Authorization.
After successful upload a job is initiated via Amazon SQS (https://aws.amazon.com/sqs/ ). This job initiates an internal worker in an independent security group responsible for picking up data and storing data in Keboola. There is no direct access possible from the internet to any of the Keboola components.
Keboola is operated in 2 Amazon availability zones in a dedicated virtual private environment (Amazon VPC: https://aws.amazon.com/vpc/ ) with hardened security framework of the Keboola application. Every Keboola component is located in a separate subnet with strict access control restrictions to protocol, service ports and source IP addresses, additionally separated into security groups within a subnet.
All data in transit and at rest are always encrypted. Data stored in Keboola reside in Amazon RDS, or optionally in Amazon Redshift. All data that occur in Amazon S3 (each and every import & export of data) is automatically archived to Amazon Glacier after certain time period (default is 180 days). All databases are backed up as well. In Amazon RDS or Amazon Redshift a snapshot is taken several times a day (currently 8 times a day). Snapshots are stored for 14 days.
Outgoing data are exported by writers - components responsible for upload of data for other data consumers. Upload is again performed via HTTPS connection.
Recipes (3rd party applications) run in Keboola environment within dockers and share the same security boundaries.
Keboola development environment is fully independent of the production environment.

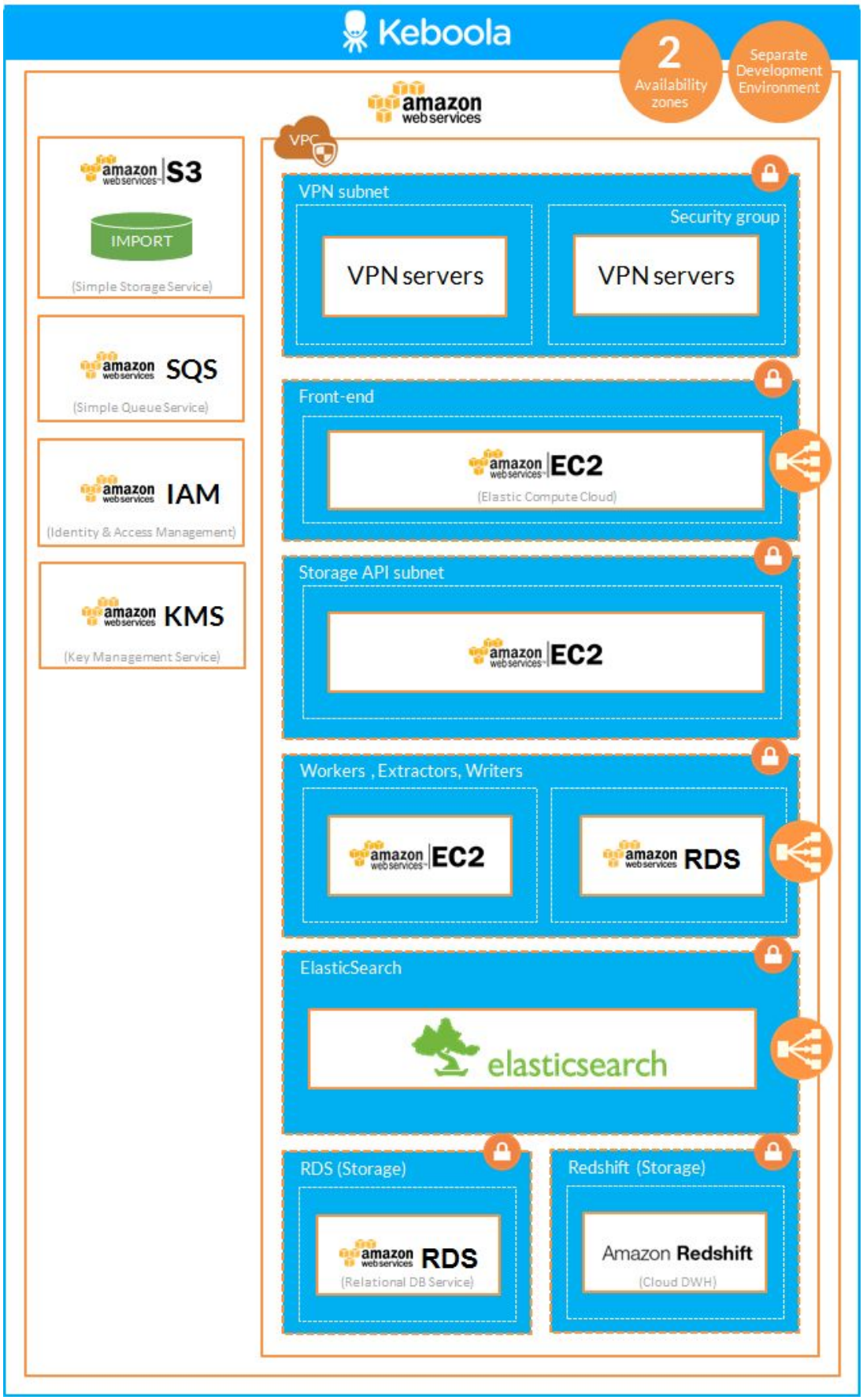



Figure 3: Keboola components & structure

## Security Concerns

### Physical Security

Keboola leverages sophisticated AWS cloud security infrastructure that has been architected to be one of the most flexible and secure cloud computing environments available today. Amazon AWS is for three subsequent years by far number 1 cloud provider on the market. Keboola runs in AWS's highly secure data centers, which utilize state-of-the art electronic surveillance and multi-factor access control systems. Data centers are staffed 24x7 by trained security guards, and access is authorized strictly on a least privileged basis. All personnel must be screened when leaving areas that contain customer data. Environmental systems in the datacenters are designed to minimize the impact of disruptions to operations, and multiple geographic regions and Availability Zones allow you to remain resilient in the face of most failure modes, including natural disasters or system failures.

### Availability & Failover

Keboola depends on availability of Amazon services. Amazon in case of EC2, EBS, RDS and Redshift claims to “use commercially reasonable efforts to make each available with a Monthly Uptime Percentage (defined below) of at least 99.95%” (https://aws.amazon.com/ec2/sla/, https://aws.amazon.com/rds/sla/ ). More expressive are hard statistics of availability at https://cloudharmony.com/status-1year where all services met 99.99+ availability.
Keboola operates in two availability zones and uses load balancing across availability zones to increase performance and availability. In case of a component failure, load balancers route traffic to the other availability zone.

### Data Location

As with other AWS services, customers can choose exact location (called “region”) of their data and Amazon guarantees that data never leave this location (they can only move within availability zones within the region). By default this is set to AWS US-East for all customers.

### Data security & Encryptio

In transit
All transport channels go through HTTPS protocol with the latest security policies of AWS (https://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-security-policy-table.html)
At rest
All data in Keboola regardless location (S3 storage, RDS or Redshift) are all encrypted. Keys are stored and managed by AWS Key Management Service (https://aws.amazon.com/kms/ ), optionally CloudHSM (https://aws.amazon.com/cloudhsm/ ) is available for additional fees.
Customers can verify grade of security configuration Keboola:
https://www.ssllabs.com/ssltest/analyze.html?d=connection.keboola.com

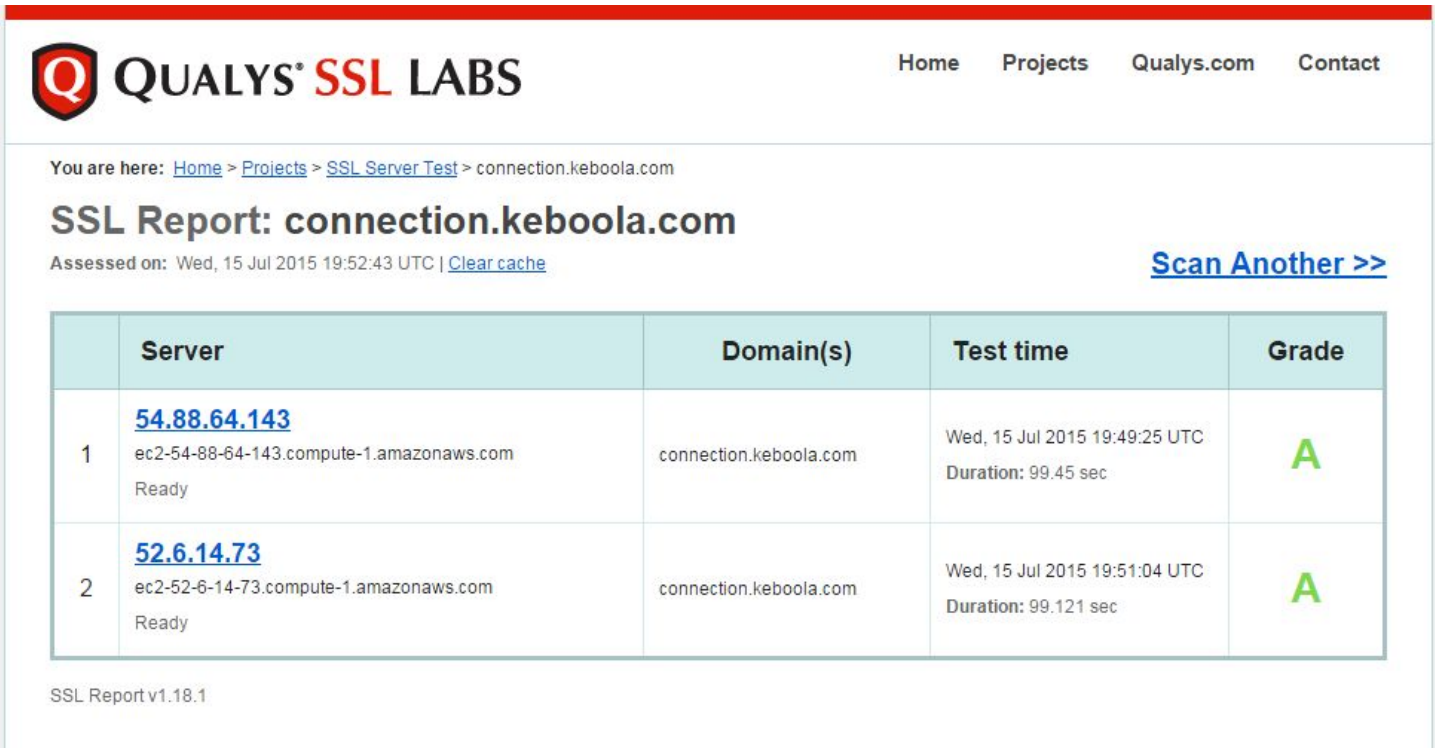
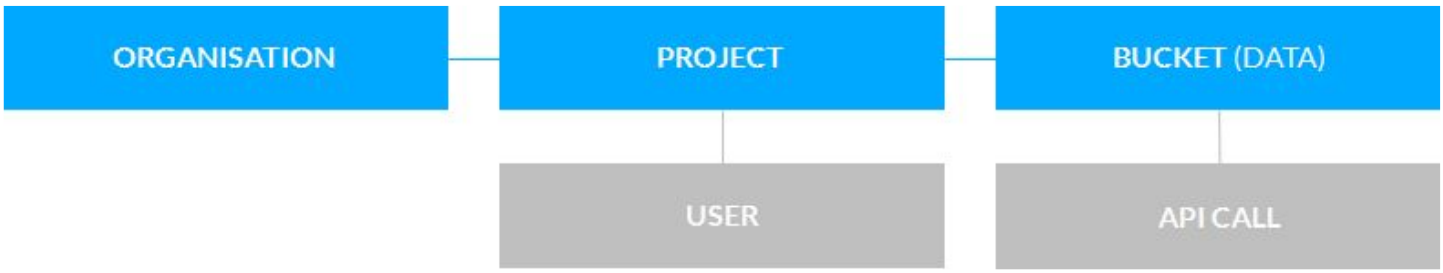



Figure 4: Verification of connection security configuration

## Authentication & Authorization

Access control within Keboola is based on Amazon IAM and uses two-phase authentication to Keboola. All communication goes through SSL connections calling REST APIs. Connection can be established only with a valid token encrypted with blowfish algorithm provided in HTTP header. Token authorization is performed by Storage API isolated from the rest of the infrastructure. End users can access user interface with login/password credentials.

Authorization model




Top level access management entity is Organisation, typically this denotes a single customer. Administrator(s) have right to manage projects within the organization. Every user is assigned to a project, on the level of projects all users are equal. On the technical level, the access control is much more fine grained. Administrator can create tokens that enable access only to a particular set of data (bucket). Your can also limit read/write operation. Buckets are defined by the customer's administrator.

Password management

Keboola enforces the following password management rules:

- minimum 8 characters long password
- 10 wrong attempts within 5 minutes enforces CAPTCHA
- all passwords are user defined (automatic passwords are not generated)
- passwords are hashed with CRYPT_BLOWFISH algorithm with automatically generated salt

- password reset is managed by emailing link to password reset

In development is two-phase authentication via Google Auth.

### Separation of environments

KBC uses Amazon AWS VPC (https://aws.amazon.com/vpc/), where all components are isolated to independent network subnets with own ACLs and routing tables. See schema in chapter Security Architecture.
Development environment is fully independent of the production environment, to the level of different Amazon region.

### Separation of roles

In Keboola is strictly enforced separation of roles.

- User roles have access to Keboola services only, all systems configurations (in Amazon console) are restricted to administrators.
- Every API call (i.e. access to functionality or data) is audited (see Auditing)
- Access to data is granted/restricted by organization's administrator (see Authorization model) to the level of a single bucket (set of data, e.g. a table)

### Client segregation

The following applies to ensure that client access to other client environments is segregated for processing and backu

- In Keboola user access permissions are assigned per project. User access to other projects and projects of other clients are restricted. Users are separated by API tokens. Everything is realised as an REST API call, no matter what API client is used (UI, C#, PHP, etc.). Every token has own access rights. All API actions are logged in realtime to isolated elasticsearch cluster and to 3rd provider (papertrailapp.com).
- In AWS access to Amazon endpoints is limited workers, isolated in AWS VPC subnet. Access to clients data enforces use of API token, validated against metadata database where access rights are evaluated.
- On the database level every customer has a dedicated Redshift database. Amazon RDS is a cheaper shared database option.
- On the S3 storage level (import/export area) every customer has a dedicated folder. Folders are not accessible directly, API manages access to the folders based on tokens in the API call.

### Data loss prevention

All data payloads are stored in Amazon AWS S3 with 99.999999999% durability. See "Data Protection" here https://aws.amazon.com/s3/faqs/ for more details and https://aws.amazon.com/s3/sla/ for SLA description

### Data backup & archiving

Data backups are stored in Amazon S3 storage with archive in Amazon Glacier handled by automatic S3 life-cycle managemet (see details:https://aws.amazon.com/s3/details/ ). Server instances are not backed up - no data are stored in application itself.
Backend databases are backuped by automatic snapshot functionality (https://aws.amazon.com/rds/faqs/#automated-backups-database-snapshots ). A snapshot of data is taken ca 8 times a day and is available for 14 days. The same applies to Amazon Redshift, which is second supported data warehouse technology -https://aws.amazon.com/redshift/. Redshift allows access to

backup via standard ODBC driver. Another way of retrieving data from Redshift is to use Keboola snapshotting API which allows customer to get Redshift tables snapshotted from Amazon S3.
All backups are also encrypted at rest.
Note: as Keboola works only as a “tunnel” for data and there are no primary operational data (only data replicas flow through Keboola), Keboola does not require data archiving.

### Physical media management

All data copies are handled by native Amazon AWS functions. Amazon AWS uses Guidelines for Media Sanitization (NIST 800-88 or DoD 5220.22-M) where all physical devices are destroyed in Amazon premises and no storage can leave Amazon premises. Detailed description can be found at the AWS Security Whitepaper page 8, paragraph “Storage Device Decommissioning”.

### Incident management

All incidents (including identified bugs) are reported immediately at https://status.keboola.com available to all customers. Customer can register to an RSS feed or subscribe to updates by email.

### Access Audit

Keboola is fully audited on the level of Amazon services. Every operation in Amazon services is executed via Amazon API. All API calls (including any operation performed via administration console of Amazon services) are logged in S3 and available to the customer. These audit logs cannot be altered by Keboola (such an activity would be again logged in the audit trail by Amazon service). Customers have full overview all operations performed over their data.
On the application level Keboola logs all API calls as events in elasticsearch and provides customers with full text search capability. Audit trail is available also programatically via an API (https://docs.keboola.apiary.io/#events ).

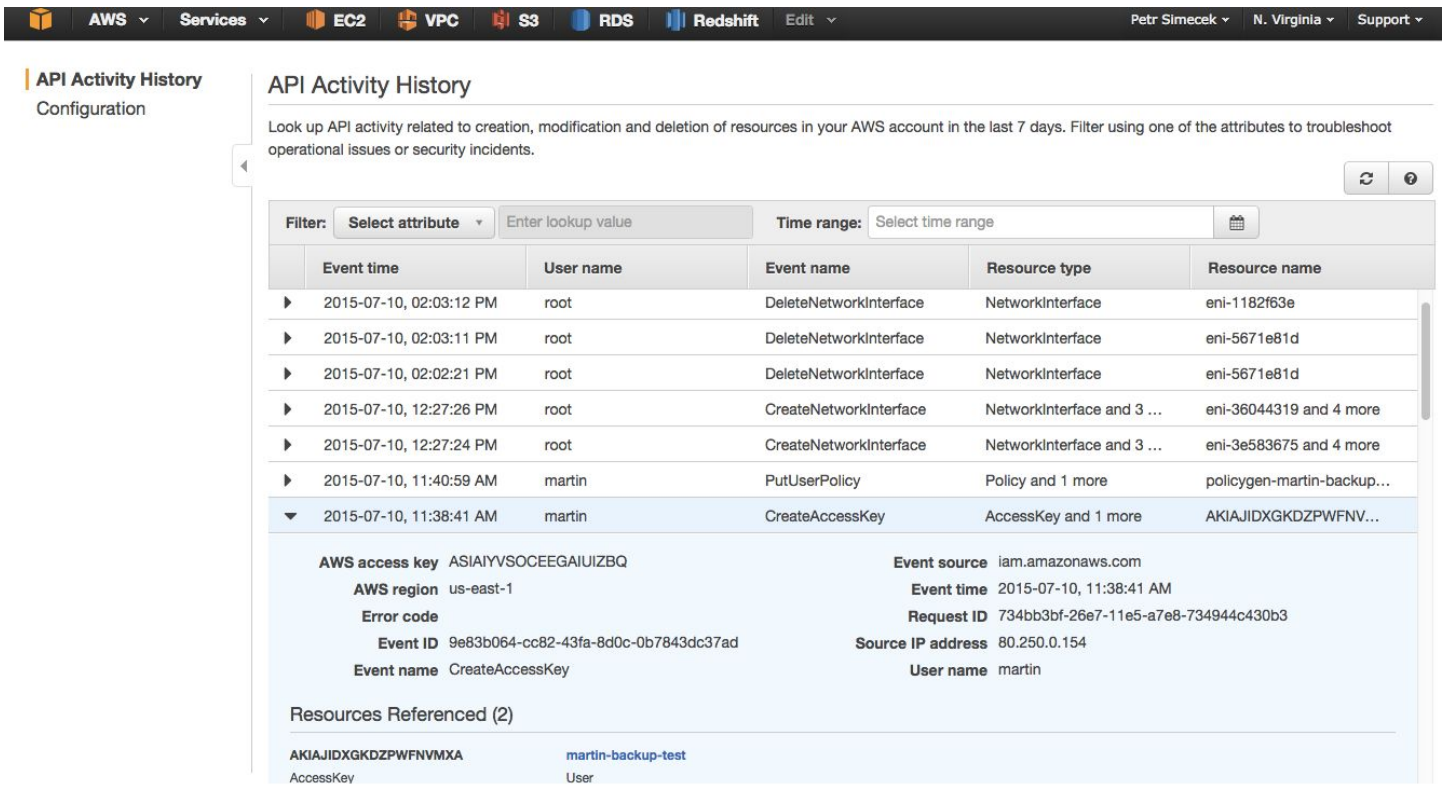



Figure 5: Real screenshot from Amazon CloudTrail for Keboola

## Monitoring

All Keboola components are continuously monitored. System availability is monitored by pingdom (https://www.pingdom.com ). Application monitoring is realized by NewRelic (https://newrelic.com/ ), this includes all API calls, SLAs, and overall application performance monitoring. All logs are processed by papertrail (https://papertrailapp.com/ ) that enables fulltext log analysis. All logs are stored for 14 days. Alerts are managed and escalated via pagerduty (https://www.pagerduty.com/ ).
Communication among all monitoring components is encrypted.

## Contract termination & Data deletion

After cancellation of the contract, all customer projects are marked as deleted. From then on customer can not access data anymore. Standard lifecycle management keeps data for 30 days. After that, objects are moved to Amazon Glacier, where they stay archived forever. All customer data can be removed fully and immediately upon customer request.

## Certifications & Compliance

- Underlying Amazon AWS is dedicated to comply with the most demanding certifications, namely:
- Sarbanes-Oxley (SOX) compliance
- ISO 27001 Certification
- PCI DSS Level I Certification
- HIPAA compliant architecture
- SOC1 Audit, SOC2, SOC3
- FISMA MediumATO
- Service Health Dashboard

For full up-to-date list of certifications and compliance audit reports seehttps://aws.amazon.com/compliance/. As a reference of used Amazon service please refer to chapter Security Architecture.