Security & Risk Management

Program Overview

Last Update: September 2018

Table of Contents

1. OUR COMPANY AND PRODUCT

Vivocha is an emerging provider of Next Generation Customer Engagement solutions. Its award-winning platform enables businesses to seamlessly communicate with prospects and customers right on the website or Mobile App, using any combination of Video, Voice, Chat, and collaboration tools like assisted browsing, form & document sharing. A sophisticated proactive engine optimises contacts to reduce service cost and avoid redundant calls to the contact center.

More than 150 companies around the world, including INGDirect, DHL, Crédit Agricole, L'Occitane, AXA, Genertel, Allianz, TUI, Engie, National Bank Of Kuwait, Hastings Direct, E.ON, Accenture, NewLook, NTT Data, Postcode lottery, trust Vivocha' technology to improve their Online customer care processes.

As a stand-alone platform, or fully integrated with pre-existing contact center technologies, Vivocha dramatically reduces deployment time and integration cost, resulting in fast ROI.

2. VIVOCHA'S SECURITY AND RISK GOVERNANCE

We operate a cloud-based digital platform—Software as a Service or SaaS—for you to engage your customers online using the most effective communication channel at the right time. We call it the Vivocha Services. We're committed to building reliable and secure systems that you can depend on for your business.

Security, privacy and compliance policies are some of the most common areas for questions we receive about Vivocha.com cloud service. Organisations using Vivovha.com are concerned about the safety of their data and that access to their data is reliable. This document aims to answer many of the frequently asked questions by IT security staff on these topics when they are considering vivocha.com.

Rest assured that Vivocha uses advanced security technology to maximise the safety and security of your company information. Vivocha implements strict security controls aimed at protecting each user, application and system, with the confidentiality of your data and the integrity of your business as top priorities.

3. OUR SECURITY AND RISK MANAGEMENT OBJECTIVES

Vivocha's primary security focus is to safeguard our customers' data. This is the reason that Vivocha has invested in the appropriate resources and controls to protect and service our customers. This investment includes the implementation of the dedicated Security Team. The Security Team is responsible for the Vivocha's comprehensive security and risk management program and the governance process. The security team is focused on defining new and refining existing controls, implementing and managing the Vivocha security framework as well as providing a support structure to facilitate effective risk management.

We have developed our security framework using best practices in the SaaS industry. Our key objectives include:

- Customer Trust and Protection – consistently deliver superior product and service to our customers while protecting the privacy and confidentiality of their information.
- Availability and Continuity of Service – ensure ongoing availability of the service and data to all authorised individuals and proactively minimise the security risks threatening service continuity
- Information and Service Integrity – ensure that customer information is never corrupted or altered inappropriately.
- Compliance with Standards – implement process and controls to align with current international regulatory and industry best practice guidance. We have designed our security program around best-of-breed guidelines for cloud security.

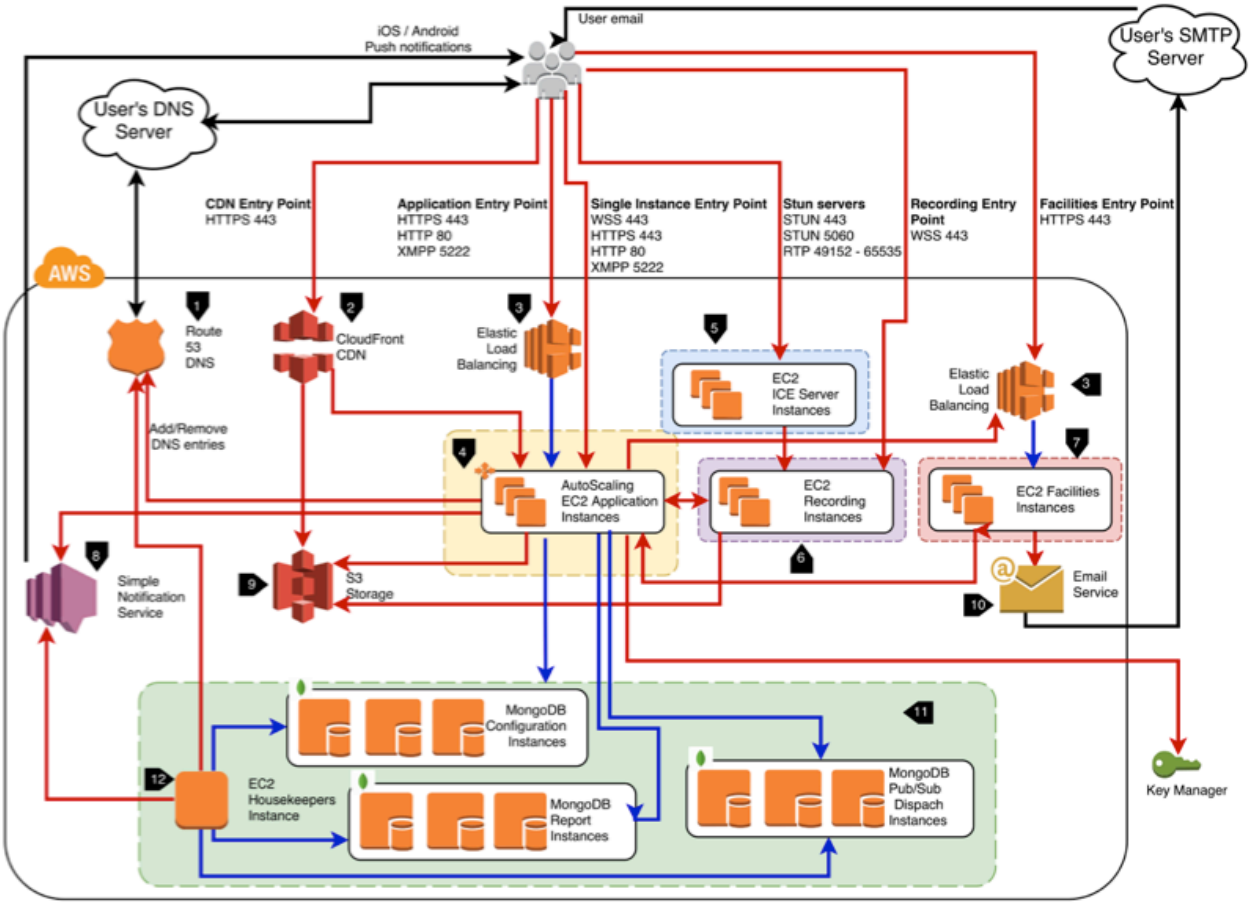
4. VIVOCHA PLATFORM ARCHITECTURE, SECURITY CONTROLS AND CAPABILITIES

Vivocha is a software company and a services company. Security is layered into the application but physical security is just as important.

In order to ensure we protect data entrusted to us, we implemented an array of security controls. Vivocha's security controls are designed to allow for a high level of employee efficiency without artificial roadblocks, while minimising risk. The following sections describe a subset of controls.

Platform Architecture

Vivocha is a scalable and highly available cloud based platform build on top of **Amazon Web Services (AWS)**. Vivocha is deployed onto different **availability Zones (AZs)**, distinct geographic locations that are engineered to insulate against failures in other AZs. Multiple AZs are combined into different **Regions**. Vivocha web applications are deployed in different AZs and different regions to ensure high availability.



1. The user's DNS requests are served by **Amazon Route 53**, a highly available Domain Name System (DNS) service. Network traffic is routed to Vivocha infrastructure running in **Amazon Web Services**.
2. To make the content delivery faster, part of the static assets is served via **Amazon Cloud Front**, the content delivery network deeply integrated and optimised to work with popular AWS services, optimised for low latency and high data transfer speeds.
3. HTTP requests are first handled by **Elastic Load Balancing** which automatically distributes incoming application traffic across multiple EC2 instances across Availability Zones.
4. Vivocha Application Servers (**worlds**) are deployed on Amazon EC2 instances. With **Auto Scaling** we can ensure that the number of EC2 instances we're using increases seamlessly during demand spike to maintain performances.
5. For multimedia communications (audio/video), there are specific instances implementing ICE, **Interactive Connectivity Establishment**, a technique used in computer networking to find ways for two computers to talk to each other as directly as possible in peer-to-peer networking, where devices usually live behind one or more level of NATs and enterprise firewalls.
6. Audio/Video communications can be recorded on our **EC2 Recording** instances
7. The **facility servers** must relieve the application servers from secondary tasks, such as the web site thumbnails generation.
8. **Amazon Simple Notification Service (Amazon SNS)** is a flexible, fully managed pub/sub messaging and mobile notifications service for coordinating the delivery of messages to subscribing endpoints and clients.
9. Resources and static contents used by the Vivocha applications are stored on **Amazon Simple Storage Service (Amazon S3)**, a highly durable storage infrastructure designed for mission-critical and primary data storage.
10. Email notifications are delivered through **Amazon Simple Email Service (Amazon SES)**, a cloud-based email sending service running on the highly reliable Amazon Web Services infrastructure.
11. **Database replication** ensures redundancy, backup, and automatic failover. A three-member replica sets provide enough redundancy to survive most network partitions and other system failures. Additionally, these sets have sufficient capacity for many distributed read operations.
12. The housekeepers help to keep database consistency.

Security Framework

Vivocha.com leverages Amazon Web Services (AWS) for our computing infrastructure. AWS has achieved **ISO 27001 certification** and and has been validated as a Level 1 service provider under the Payment Card Industry (PCI) Data Security Standard (DSS). They undergo annual SOC 1 audits and have been successfully evaluated at the Moderate level for Federal government systems as well as DIACAP Level 2 for DoD systems.

For more detail on AWS security, please refer to http://aws.amazon.com/security/

Physical Security of Facilities

Vivocha.com employees do not have physical access of any kind to our production facilities, as all of our infrastructure is in the cloud at AWS.

Data centers are staffed 24x7 by trained security guards, and access is authorised strictly on a least privileged basis. Environmental systems are designed to minimise the impact of disruptions to operations. And multiple geographic regions and Availability Zones allow to remain resilient in the face of most failure modes, including natural disasters or system failures.

Our Infrastructure provider - Amazon - has many years of experience in designing, constructing, and operating large-scale data centers. This experience has been applied to the AWS platform and infrastructure. AWS data centers are housed in nondescript facilities, and critical locations have extensive setback and military grade perimeter control berms as well as other natural boundary protection. Authorised staff must pass two-factor authentication no fewer than three times to access Amazon Web Services Security data center floors. All visitors and contractors are required to present identification and are signed in and continually escorted by authorised staff.

Amazon only provides data center access and information to employees who have a legitimate business need for such privileges. When an employee no longer has a business need for these privileges, his or her access is immediately revoked, even if they continue to be an employee of Amazon or Amazon Web Services. All physical and electronic access to data centers by Amazon employees is logged and audited routinely.

Network Security

The Amazon EC2 inbound firewall is configured in a default deny mode and Vivocha.com explicitly opens ports to allow inbound traffic. The traffic may be restricted by protocol, by service port, as well as by source IP address (individual IP or CIDR block).

The firewall is configured to permit only the absolute minimum connectivity required to provide the Vivocha.com services. Changes to firewall access rules require Vivocha.com's X. 509 certificate and key for authorisation.

The AWS network provides significant protection against traditional network security issues. The following are a few examples:

- **Distributed Denial Of Service (DDoS) Attacks:** AWS API endpoints are hosted on the same Internet-scale, world class infrastructure that supports the Amazon.com retail site. Standard DDoS mitigation techniques such as syn cookies and connection limiting are used. To further mitigate the effect of potential DDoS attacks, Amazon maintains internal bandwidth which exceeds its provider-supplied Internet bandwidth.

- **Man In the Middle (MITM) Attacks:** All of the AWS APIs are available via SSL-protected endpoints which provides server authentication. Amazon

EC2 AMIs automatically generate new SSH host keys on first boot and log them to the console. Vivocha.com then uses the secure APIs to call the console and access the host keys before logging into the instance for the first time.

- **IP Spoofing:** Amazon EC2 instances cannot send spoofed traffic. The Amazon-controlled, host-based firewall infrastructure will not permit an instance to send traffic with a source IP or MAC address other than its own.

- **Port Scanning:** Port scans by Amazon EC2 customers are a violation of the Amazon EC2 Acceptable Use Policy (AUP). Violations of the AUP are taken seriously, and every reported violation is investigated. When Port scanning is detected it is stopped and blocked. Port scans of Amazon EC2 instances are generally ineffective because, by default, all inbound ports on Amazon EC2 instances are closed.

- **Packet sniffing by other tenants:** It is not possible for a virtual instance running in promiscuous mode to receive or "sniff" traffic that is intended for a different virtual instance. The hypervisor will not deliver any traffic to instances that is not addressed to them. This includes two virtual instances that are owned by the same customer, even if they are located on the same physical host. Attacks such as ARP cache poisoning do not work within EC2.
  The Operations Team has the ability to change firewall rules.

Host Security

We are leveraging AWS for all of our computing infrastructure. AWS owns the physical hardware. AWS provides security groups to limit access to devices. We fully utilise security groups to limit access to our computing resources.

Our production environment is completely separate from the other environments, including development and QA. The development and QA environments are in the west-2 region (Oregon, USA), the production environments are currently in the east-1 (North Virginia, USA) and eu-1 (Ireland, Europe) regions.

AWS provides Identity Access Management (IAM) to control access to AWS resources. We use AWS IAM to manage separate, restrictive AWS credentials for each of our environments. This limits the AWS services available to each environment and compartmentalises them.

We also use AWS IAM to delegate monitoring and management capabilities to operations staff and prevent destructive actions.

SSH keys are required to gain console access to our servers, in any of the environments. Individually identifiable RSA key pairs are used for SSH access, and root login is disabled. This insures that there is a complete audit trail via sudo from a specific action back to the specific individual who triggered that action.

We adhere to strong password policies and require that all RSA private keys be encrypted with a compliant password.

Automated processes are in place on each host that monitoring for unauthorised login attempts, with the offending IP address being automatically blacklisted and an alert being generated.

Hardened Operating System

Vivocha runs on hardened Linux servers. Externally exposed critical patches are addressed within 24 hours.

Internal and Third Party Testing

Vivocha routinely runs internal and external vulnerability scans and penetration tests. Third party firms are utilised to perform in-depth quarterly security reviews.

Password Policies

Administrative controls allow the definition of password policies for length, expiry, and complexity to mirror your corporate password policies.

Business Continuity

Your data is backed up multiple times a day. Backups are transferred offsite over SSH and properly deleted after six months.
The servers are built using repeatable build processes. All changes to the production environment pass through a peer-review change management process, with all changes logged to a central ticket system.

Secure Connections

All connections to Vivocha.com session controlled APIs are secured via SSL/TLS. Any attempt to connect over HTTP is redirected to HTTPS.

Application Security

Vivocha utilises secure development best practices that integrate security reviews throughout design, prototype, and deployment.

Encryption

We have implemented strong encryption via SSL in our application. By using encryption, we minimise the chances of someone possibly intercepting username/password combinations and/or other sensitive information.

Areas where we utilise SSL include:

- All application logins require SSL. Any area which requires a user to log into our system also requires that SSL is used.

- The administrative, agent, analytics and API interfaces all leverage and requires SSL throughout.

- Any communication session through Vivocha.com (chat, voice, video, co-browsing sessions) requires SSL throughout.

Some static assets (javascript libraries and css style sheets) have optional SSL access.

The encryption settings section in the Vivocha platform is made up of the following fields:

Secret Token: used to compute API signatures and used to encrypt/decrypt keys when using the Key Manager API

Chat Encryption: the encryption of the chat transcripts (exchanged messages) can be set to Transmission Only (basic security, all traffic uses HTTPS/SSL/TLS), Transmission & Storage (data is encrypted by the servers before storage) and End-To-End (data is encrypted directly by the clients and is stored already encrypted)

Data Collection Encryption: encryption of the data collected before and during a contact (includes data entered by the visitors on data collection forms, data attached to the contact via API and data edited by the agent)

Key Manager: set the Key Manager to Internal (the default) or External

Key Manager URL: the URL of the customer supplied Key Manager, when the Key Manager mode is External. Keys are always transmitted encrypted with the secret token, even when using HTTP. Nevertheless, for better protection, we strongly advise to use only HTTPS URLs.

Brute Force Attack Prevention

In order to minimise brute force login attacks, we automatically disable accounts for a five- minute period after five consecutive failed attempts have been registered. If we ever determine that this is a possible area of concern, we can easily increase the lockout period or decrease the number of consecutive failures via configuration.

XSS

All communication messages (user chat text and API generated chat events) and user provided data are sanitised to prevent XSS.

CSRF

Vivocha.com requires authenticity tokens in all sensitive JSON requests with corresponding verification on inbound requests.

SQL Injection

Vivocha.com is immune to SQL injection attacks, as we don't use any form of SQL in our systems.
Attachments are saved to a generated GUID temp file before uploading to S3. This avoids issues associated with saving/overwriting files with relative file paths.

Passwords

All user passwords are hashed using the MD5 algorithm with salt. Hashing passwords is actually more secure than encrypting them, because that means we don't have access to the original passwords, nor does anyone else. So even if our database is compromised, everyone's passwords will stay secure.

Complex passwords with a minimum password length of 8 characters are required by default.

Data Storage & Retention Policies

Data is generally stored in a MongoDB Database. File attachments are stored within S3. All data (other than passwords and authentication strings) is stored in clear text.

We support optional end-to-end encryption of communication messages (chat transcripts), via our APIs: in this scenario, Vivocha.com does not store the encryption key used, and has no means of decrypting the stored data.

The production MongoDB database is configured with high availability with data replicated to multiple, redundant instances. The database is backed up on a nightly basis with encrypted backup copies being shipped to secure offsite storage.

Vivocha.com is a multi-tenant SaaS solution: customer data is co-mingled on the same database collections, but all data is scoped by an account ID to ensure that one account cannot access data of another account. Unit, functional, and integration tests are run continuously on our servers to ensure that it‘s not possible for account data to leak.

In addition to our usage of this data in production we also occasionally take a copy of the data and load it in our testing environments. These copies are scrubbed of any sensitive or personally-identifiable information before being used for testing or development purposes.

Access Controls to Service Data

Having contracted with Vivocha for the service you provide Vivocha with access to your production, development or test environment, which may include personal information about your employees, customers, partners or suppliers (collectively "end users").

Vivocha's access to services data is based on job role/responsibility. Vivocha will safeguard data you send to our organisation in the same manner in which we protect our own similar confidential information.

Vivocha.com staff does not access or interact with customer data or applications as part of normal operations. There may be cases where Vivocha.com is requested to interact with customer data or applications at the request of the customer for support purposes or where required by law.

Below are the conditions under which Vivocha may access, collect and/or use services data.

- **To Provide Services:** services data may be accessed and used to fulfill the requirements specified in your order for support, consulting, or other services.

- **To Maintain and Upgrade a System:** Technical staff may require periodic access to services data to monitor system performance, test systems and develop and implement upgrades to systems. Any temporary copies of services data created as a necessary part of this process are only maintained for time periods relevant to those purposes.

- **To Address Performance and Fix Issues:** On occasion, Vivocha may develop new versions, patches, updates, and other fixes to its programs and services (such as security patches addressing newly discovered vulnerabilities). In accordance with the terms of your order for services and/or with notice to you, we may access and/or use a copy of your test, development or production environment, including services data, to test such new versions, patches, updates and fixes and validate that they work in your environment(s).

- **As a Result of Legal Requirements:** Vivocha may be required to provide personally identifiable information to comply with legally mandated reporting, disclosure or other legal process requirements.

Vivocha may transfer and access services data globally as required for the purposes specified above. If Vivocha hires subcontractors to assist in providing services, their access to services data will be consistent with the terms of your order for services and this services privacy policy.

Vivocha does not use services data except as stated above or in your order. Vivocha may process services data, but does not control your information collection or use practices for services data. If you provide any services data to Vivocha, you are responsible for providing any notices and/or obtaining any consents necessary for Vivocha to access and use services data as specified in this policy and your order.

Each Vivocha Employee and subcontractor must sign a confidentiality agreement, specifically covering access to services data.

Penetration Testing

Vivocha.com strives to provide a robust and trustworthy service for our customers. We take security very seriously and continually monitor our services for suspected attack. We also understand that security is a partnership between us and our customers. A critical phase of any secure application deployment involves testing applications for potential vulnerabilities.

Our Terms and Conditions (http://www.vivocha.com/tos) describes permitted and prohibited behaviour on Vivocha.com and includes descriptions of prohibited security violations and network abuse. However, because penetration testing frequently is indistinguishable from these activities, we have established a policy for customers to request permission to conduct penetration tests: please send an email to support@vivocha.com, providing as much details as possible, including the nature of the tests that you want to perform, where the tests would be performed from and who will be conducting them, and contact details of who will be responsible for them on your side. Our staff get in contact with you as soon as possible to discuss you request and eventually grant you our authorisation.

No further action on your part is required after you receive our authorisation. You may conduct your testing through the conclusion of the period agreed. If you need more time for additional testing, contact our support staff again asking to extend your test period to the new date. You are not authorised for an extension unless you receive a new authorisation from us.

Password policy

Setting several parameters for the password policy is possible with Vivocha. Those will apply to all the new accounts users being created or edited from that moment on. The parameters that can be set are listed here below:

- password maximum length
- forbid user id within password field
- lowercase letter
- uppercase letter
- digit
- special character

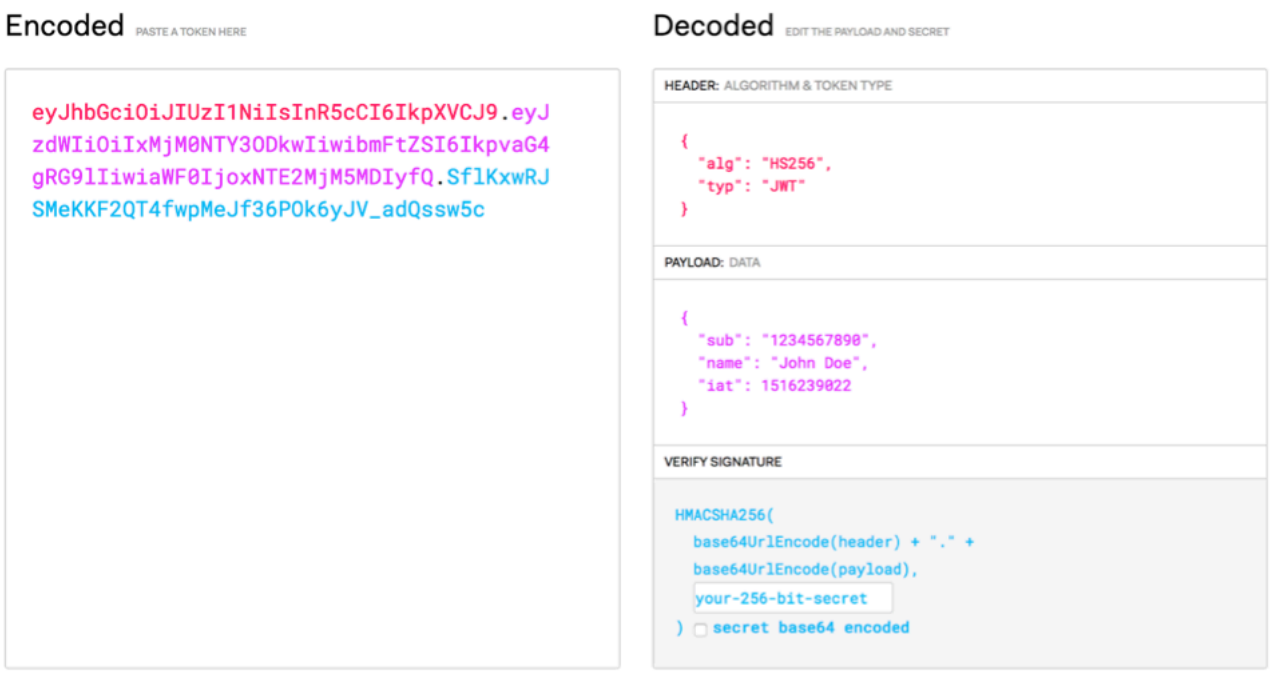
Access Restriction

It is possible to restrict the access to the Vivocha console by defining a range of authorised IP addresses, using the CIDR-v4 format.

JWT security high level solution design

This paragraph is meant to provide an high level design of the architectural elements that would be necessary to implement JWT token authorisation mechanism among the different actors.

JWT introduction: JSON Web Token (JWT) is an open standard (RFC 7519) that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret (with the HMAC algorithm) or a public/private key pair using RSA.
Although JWTs can be encrypted to also provide secrecy between parties, we will focus on signed tokens. Signed tokens can verify the integrity of the claims contained within it, while encrypted tokens hide those claims from other parties. When tokens are signed using public/private key pairs, the signature also certifies that only the party holding the private key is the one that signed it.
(please refer to https://www.jwt.io)



Activities

A “shared secret” must be agreed among the parties. The shared secret is essential for assuring authenticity of the communication. The shared secret will be used to forge the

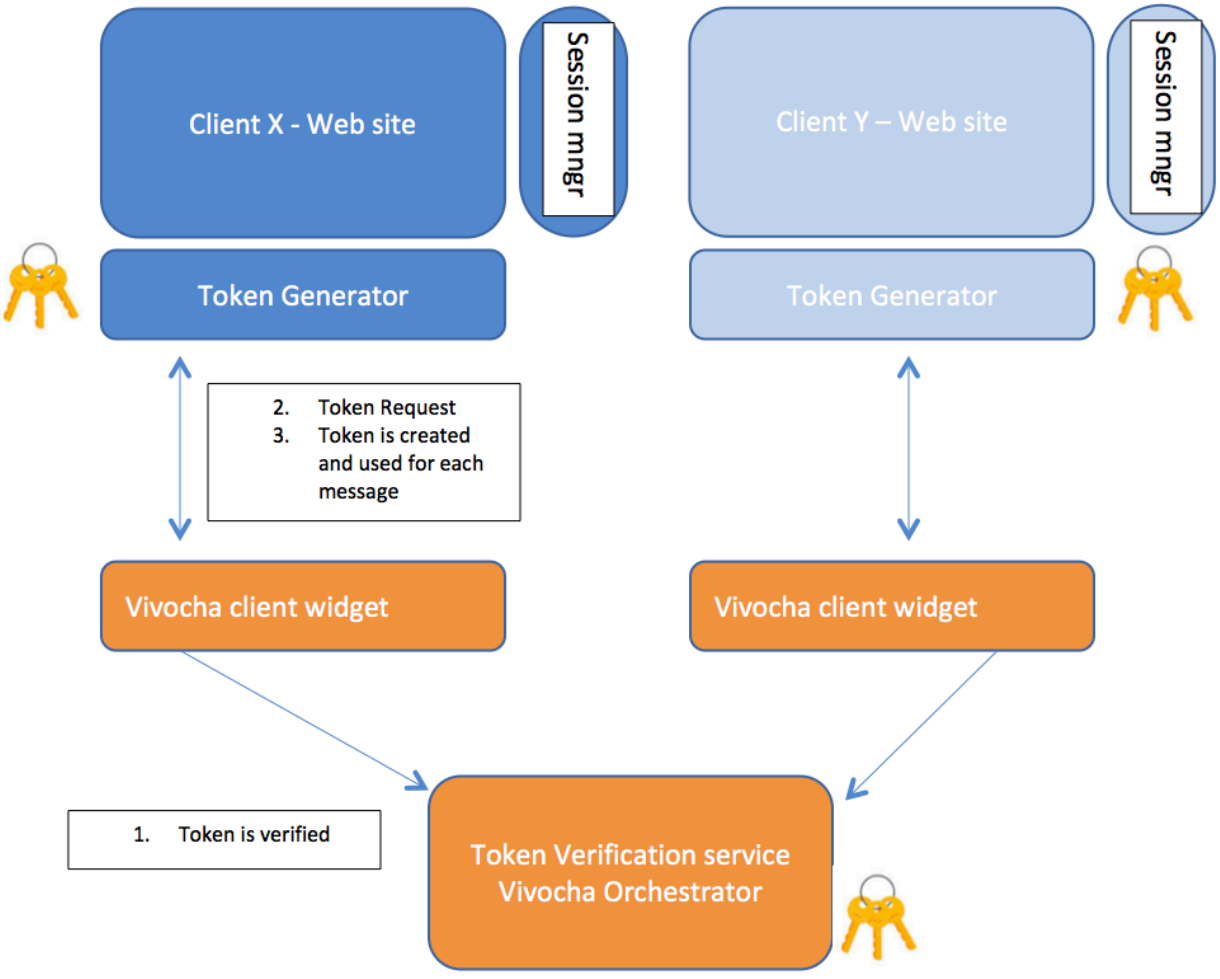
signature (typically created using hash algorithm using part of the payload and the shared secret).

The JWT token format, the request messages, the response will be formally agreed so that the structure will be the same for all customers.

The secret must be shared with all the actors that need to have authorisation for communication.

Even if the request, response, and token structure will be agreed and standard for all the customers. It is necessary for each customer to implement the service that will generate the tokens. This service and the logic for granting the token is local for each customer and the technology may vary from customer to customer.

Here below an high-level representation of the architecture (the keys represent the shared secret availability):



5. DOCUMENT SCOPE AND USE

Vivocha values transparency in the ways we provide solutions to our customers. This document is designed with that transparency in mind. We are continuously improving the protections that have been implemented and, along those lines, the information and data in this document (including any related communications) are not intended to create a binding or contractual obligation between Vivocha and any parties, or to amend, alter or revise any existing agreements between the parties.