HAVENTEC SANCTUM

INTRODUCTION

We are living in an era of truly remarkable change. Industries and enterprises are being transformed by new technologies. Incumbents are being disrupted. Disruptors are redefining markets and transforming customer expectations. With more than half the world's population now online, organisations seeking to remain relevant are relying on digital technologies to underpin access to, and meaningful engagement with, their markets. This rapidly expanding connectivity is in turn amplifying the impact of bad actors. The increasing sophistication and frequency of cyber-attacks, as well as the increasing impact of basic human error, are heightening concerns about privacy. Violations of privacy are driving a new focus from regulators around the world, each mandating new rules and data rights to both protect consumers and sanction increasing criminal penalties for data breaches.

Global economies have been able to thrive based on the existence of trust between transactors in an exchange. The future of open government and the open enterprise is heavily dependent on the existence of digital trust. However, current technologies are providing inadequate protection to the privacy, misuse, or theft of sensitive data. This is the acute and unresolved challenge facing the digital world. The core digital stakeholders (consumers / enterprises / regulators) are all seeking a solution to the inherent weaknesses of the current digital trust model. Addressing the weaknesses will require redesigning the foundations of digital trust which involves:

1. revisiting the need to centrally store sensitive data;
2. changing the way applications interact with that data;
3. changing the way individuals, the enterprise and third-parties access that data; and
4. providing individuals with more control over how their personal information is used and shared.

At the core of the Haventec proposition is a remarkably simple idea, beautifully executed – protecting digital identities from harm, and keeping 'data that matters' safe – the key foundations for digital trust. Haventec have developed a suite of technologies that eliminate the need to centrally store sensitive credentials and data, providing the ultimate preparedness for when the enterprise is breached. Due to the decentralised nature of these innovative technologies, the sensitive data and credentials currently stored centrally are not there.

Haventec delivers a haven for sensitive credentials and data.

1. For consumers and citizens, it provides them the capacity to extract value from their personal digital transaction data while maintaining their individual right of privacy.

2. For governments and regulators, it provides a credible alternative to the current approaches adopted for protecting the sensitive data of individuals, promoting the safeguarding of the broader interests of society by sanction for mass data and credential breach.

3. For digital suppliers and data custodians, it allows them to extend controls to their customers, placing them in charge of how their data is shared and used. It eliminates the risk of mass credential and data breaches and provides additional controls to the enterprise to better manage regulation of individual privacy that is in the best interests of the business, its shareholders, the Boards of Directors, and their customers, thereby strengthening trust.

WHAT IS SANCTUM

Haventec's Sanctum keeps 'data that matters' safe. The platform obviates existing methodologies of storing data centrally. Sanctum securely deconstructs sensitive data into distinct components to allow for easy decentralisation. These components in isolation are of no value, but will, when combined using Sanctum, allow access to the data once again. Two sets of keys are changed and re-encrypted for every Sanctum transaction.

Sanctum provides organisations that need access to sensitive data the ability to access that data without the risk of centrally story the data. By removing sensitive information from an enterprises network, the risks of financial and reputational damage due to sensitive data theft is eliminated.

Sanctum leverages the foundational concepts of Haventec Authenticate to securely deconstruct and store distributed data puzzles across users, their devices and the network. This allows an enterprise to give control of personal data back to their customers while directly addressing business risk of sensitive data protection.

HOW DOES IT WORK?

Haventec Sanctum offers the ultimate protection for when your organisation must deal with the impact of a data breach. Specifically:

- Removing sensitive information off your network;
- Securely managing data in decentralised environments;
- Safely recalling data at any time to authorised users;
- Eliminating mass data breach risk; and
- Restoring control of personal data to the owner.

To view a video demonstration on how Haventec Sanctum works, please click on this link: https://youtu.be/_NoisEcSwJE

SANCTUM VAULTS OVERVIEW

Haventec's Sanctum decentralised data vault platform provides two types of Sanctum vault to protect your 'data that matters':

1. Sanctum online vaults secure and protect data by decentralising a cryptographic data component on to the end user's device. These types of vaults can only be transacted with when the end user, who has control of the device, is online and actively transacting with a digital experience.

2. Sanctum offline vaults secure and protect data by deconstructing an encrypted data payload into a complex cryptographic math puzzle which enables the derivation of data on-demand when an organisation needs to recall it. Sanctum offline vaults provide enhanced protections through the distribution of the math puzzle components to ensure that all required components are securely decentralised.

ONLINE VAULT PROCESSES

REGISTERING AN ONLINE VAULT

This following description outlines the technical algorithmic flow implemented in the register online vault process:

1. The user will enter a username (U_1) and unencrypted data (UD_1), e.g. a credit card number.

2. The client device will gather a set of device metadata and send that along with the user entered information.

3. The system will generate a symmetric key consisting of an initialisation vector (IV_1) and key (K_1)) (256 bit key length).

4. The system will encrypt the unencrypted data (UD_1) using the Initialisation Vector (IV_1) and key (K_1) using the AES-256-GCM algorithm to form a Base64 encoded string known as the encrypted data (or encryptedData (ED_1)).

5. The system will generate a vault UUID (ID_1) and link it to the username (U_1) and device metadata within the database store.

6. The system will link the Initialisation Vector (IV_1), key (K_1) to the vault UUID (ID_1) within the database store.

7. The system will return the vault UUID (ID_1) and encryptedData (ED_1) to the user.

8. The user's device will link the vault UUID (ID_1) and encryptedData (ED_1) to the username (U_1) within a device specific local storage.

TRANSACTING AN ONLINE VAULT

The following description outlines the technical algorithmic flow implemented in the transact online vault process:

1. The user's device will retrieve the vault UUID (ID_1) and encryptedData (ED_1) from local storage.

2. The client device will gather device metadata and send this along with the user entered data from step 1.

3. The system will validate that the device metadata is consistent with the metadata that was used during registration.

4. The system will retrieve the Initialisation Vector (IV_1) and key (K_1) based of the provided vault UUID (ID_1).

5. The system will decrypt the encryptedData (ED_1) to get the unencrypted data (UD_1) using the Initialisation Vector (IV_1) and key (K_1).

6. The system will generate a new symmetric key (Consisting of an initialisation vector (IV_2) and key (K_2)) (256 bit key length).

7. The system will encrypt the unencrypted data (UD_1) using the Initialisation Vector (IV_2) and key (K_2) using the AES-256-GCM algorithm to form a Base64 encoded string known as the encrypted data (or encryptedData (ED_2)).

8. The system will link the Initialisation Vector (IV_2) and key (K_2) to the vault UUID (ID_1) within the database store.

9. The system will return the unencrypted data (UD_1) and encryptedData (ED_2) to the user.

10. The user's device will link the encryptedData (ED_2) to the username (U_1) within a device specific local storage.

OFFLINE VAULT PROCESSES

REGISTER AN OFFLINE VAULT FOR PAYMENTS

PCI COMPLIANT ALGORITHM:

The following description outlines the technical algorithmic flow implemented in the register offline PCI vault process:

1. The user enters a username (U_1) unencrypted data (UD_1), e.g. a credit card number.

2. The system will generate a vault UUID (ID_1), prime number (P_1) of size 2048bit, initialisation vector (IV_1) and symmetric key (K_1) (256 bit key length).

3. The system will pad the unencrypted data (UD_1) with random data to produce the padded unencrypted data (PUD_1).

4. The system will encrypt the padded unencrypted data (PUD_1) with the initialisation vector (IV_1) and symmetric key (K_1) to create an encrypted data (ED_1).

5. The encrypted data (ED_1) will be converted to a large integer (LI_1) via ASCII translation table.

6. The system will calculate the next prime number (NP_1) after LI_1 and difference between the two values will be the payload (PL_1).

7. The system will calculate the multiplied primes (N_1) by multiplying P_1 and NP_1.

8. The system will store the prime number (P_1), initialisation vector (IV_1) and symmetric key (K_1) against the vault UUID (ID_1) within the database store.

9. The system will return the multiplied primes (N_1) and payload (PL_1).

10. The system will purge next prime (NP_1), payload (PL_1) and multiplied prime (N_1) from memory.

TRANSACT AN OFFLINE VAULT FOR PAYMENTS

PCI COMPLIANT ALGORITHM:

The following description outlines the technical algorithmic flow implemented in the transact offline PCI vault process:

1. The user will submit the vault UUID (ID_1), multiplied prime (N_1) and payload (PL_1).

2. The system will retrieve the prime number (P_1), initialisation vector (IV_1) and the symmetric key (K_1) via the user supplied vault UUID (ID_1).

3. The system will divide the multiplied prime N_1 by the stored prime number P_1 to calculate NP_1.

4. The system will subtract the payload PL_1 from NP_1 to calculate the large integer LI_1.

5. The large integer (LI_1) will be converted to the encrypted data (ED_1) via ASCII translation table.

6. The system will decrypt to the padded unencrypted data (PUD_1) with the symmetric key (K_1) and initialisation vector (IV_1).

7. The system will remove the padding to produce the unencrypted data (UD_1).

8. The system will run steps 1-7 of Register an offline vault for credit cards for the next transaction.

9. The unencrypted data (UD_1) will be returned to the user, along with another credit card offline vault.

REGISTER AN OFFLINE VAULT FOR GENERAL DATA

FOR DATA THAT IS <= 100 BYTES:

The following description outlines the technical algorithmic flow implemented in the register offline general data vault (<=100 bytes) process:

1. The user will enter a username (U_1), unencrypted data (UD_1) (e.g. a Base64 encoded photo).

2. The system will generate a vault UUID (ID_1), prime number (P_1) (of size 1520 bit), initialisation vector (IV_1), symmetric key (K_1) (256 bit key length).

3. The system will pad the unencrypted data (UD_1) with random data to a 100 bytes to create padded unencrypted data (PUD_1).

4. The system will encrypt the padded unencrypted data (PUD_1) with the symmetric key (K_1) and initialisation vector (IV_1) to create an encrypted data (ED_1).

5. The encrypted data (ED_1) will be converted to a large integer (LI_1) via ASCII translation table.

6. The system will use the prime number (P_1) and an algorithm to split the large integer (LI_1) into two or more parts ($SP_{1...n}$).

7. The system will store the vault UUID (ID_1), the first part (SP_1), initialisation vector (IV_1) symmetric key (K_1), prime number (P_1) within the database store.

8. The system will return the vault UUID (ID_1), the remaining parts ($SP_{2...n}$).

FOR DATA THAT IS > 100 BYTES:

The following description outlines the technical algorithmic flow implemented in the register offline general data vault (>100 bytes) process:

1. The user will enter a username (U_1) and unencrypted data (UD_1) (e.g. a photo).

2. The system will generate a vault UUID (ID_1), prime number (P_1) (size 1520 bit), two symmetric keys (K_1) (K_2) (256 bit key length) and their respective initialisation vectors (IV_1) and (IV_2).

3. The system will encrypt the unencrypted data (UD_1) with the symmetric key (K_1) and initialisation vector (IV_1) to create encrypted data (ED_1).

4. The system splits encrypted data (ED_1) into two parts taking the first 100 bytes (FBD_1) and remaining (RBD_1) (20/80 split).

5. The system will pad the symmetric key (K_1) and initialisation vector (IV_1) with random data, to create padded unencrypted key (PUK_1).

6. The system will encrypt the padded unencrypted key (PUK_1) with the symmetric key (K_2) and initialisation vector (IV_2) to create an encrypted key (EK_1).

7. The encrypted key (EK_1) will be converted to a large integer (LI_1) via ASCII translation table.

8. The system will use the prime number (P_1) and an algorithm to split the large integer (LI_1) into two or more parts ($SP_{1...n}$).

9. The system will store the vault UUID (ID_1), the first split part (SP_1) and the symmetric key (K_2), initialisation vector (IV_2), prime number (P_1) and remaining data bytes (RBD_1) within the database store.

10. The system will return the vault UUID (ID_1), the remaining parts ($SP_{2...n}$), and the first bytes of data (FBD_1).

TRANSACT AN OFFLINE VAULT FOR GENERAL DATA

FOR DATA THAT IS <= 100 BYTES:

The following description outlines the technical algorithmic flow implemented in the transact offline general data vault (<=100 bytes) process:

1. The user will submit the vault UUID (ID_1) and the parts ($SP_{2...n}$).

2. The system will retrieve the prime number (P_1), part 1 (SP_1), initialisation vector (IV_1), symmetric key (K_1) via the user supplied vault UUID (ID_1) from the database store.

3. The system will use (P_1) and the parts ($SP_{1...n}$) to rebuild the large integer (LI_1).

4. The large integer (LI_1) will be converted to the encrypted data (ED_1) via ASCII translation table.

5. The system will decrypt the encrypted data (ED_1) to the padded unencrypted data (PUD_1) with the initialisation vector (IV_1) and symmetric key (K_1).

6. The system will run steps 1-8 of Register an offline vault for generic data for the next transaction.

7. The unencrypted data (UD_1) will be returned to the user, along with another offline vault if applicable.

FOR DATA THAT IS > 100 BYTES:

The following description outlines the technical algorithmic flow implemented in the register offline general data vault (>100 bytes) process:

1. The user will submit the vault UUID (ID_1), the split parts ($SP_{2...n}$), and the first bytes of data (FBD_1).

2. The system will retrieve the prime number (P_1), split part 1 (SP_1), initialisation vector (IV_2), symmetric key (K_2) and remaining data bytes (RBD_1) via the user supplied vault UUID (ID_1).

3. The system will use (P_1) and the parts ($SP_{1...n}$) to rebuild the large integer (LI_1).

4. The large integer (LI_1) will be converted to the encrypted key (EK_1) via ASCII translation table.

5. The system will decrypt encrypted key (EK_1) to the padded unencrypted key (PUK_1) with the symmetric key (K_2) and initialisation vector (IV_2).

6. The system will remove the padding to produce the unencrypted key (K_1).

7. The system will merge the first bytes of data (FBD_1) with the remaining bytes of data (RBD_1) to create encrypted data (ED_1).

8. The system will use unencrypted key (K_1) and unencrypted initialisation vector (IV_1) on encrypted data (ED_1) to produce unencrypted data (UD_1).

9. The system will run steps 1-8 of Register an offline vault for generic data for the next transaction.

10. The unencrypted data (UD_1) will be returned to the user, along with another offline vault if applicable.

NOTABLE FEATURES

- Sanctum only stores one part of the multi-part puzzle required to authenticate the user – a set of cryptographic keys that, if stolen, are of no value.
- The system database only stores a set of symmetric keys that, if leaked, are of significantly less value to an attacker than a credentials listing.
- The device metadata consistency checks for online vaults allowing for changes within a certain threshold e.g. the addition of new fonts, the upgrade of a browser version

FREQUENTLY ASKED QUESTIONS

How does Sanctum perform under scale and load?

The Sanctum platform is currently benchmarked at a sustained load of 1,000 transactions per second with the average response being between 400 to 500 milliseconds.

How can an organisation consume and deploy the Sanctum platform APIs into their ecosystem?

Two key deployment options: 1. Haventec's fully managed SaaS offering which is hosted on AWS. 2. Alternatively, we offer organisations the option to deploy our platform into an on-premise capability of their choice, e.g. private data centre, private cloud.

Can I use Sanctum vaults for secure credit card payment transactions?

Yes, Haventec's platform comes with Payment Card Industry – Data Secure Standard (PCI-DSS) Tier 1 Service Provider accreditation.

What type of datasets can be protected with Sanctum vaults?

Haventec Sanctum can be used to protect any data. This includes, but is not limited to, identity data, medical records, financial information, insurance claims, legal action, photos and corporate transaction information – any 'data that matters' to the enterprise, to the regulators and to your customers.

How can I protect user access to online and offline vaults?

Haventec Sanctum comes with native integrations to the Haventec Authenticate APIs to provide hyper secure user access. Our platform has been designed to easily integrate and operate with common identity authorisation systems such as OpenID Connect, OAuth2, JWT and Kerberos.

TALK WITH US ABOUT SECURING YOUR SENSITIVE DATA

+61 2 8320 9488
info@haventec.com