無意味化によるデータセキュリティ

「秘密分散」

ZENMU ソリューションのご紹介

株式会社ZenmuTech

秘密分散とは？

秘密分散は、1979年にBlakley博士とShamir博士によって提唱された概念

情報を複数の分散片(シェア)に分けて守り、分散片を集めることで元の情報に戻す技術

秘密分散にはいくつかの方式があるが、代表的なものとして、しきい値分散、AONT(All or Nothing Transform)といった方法がある

秘密分散の歴史

1979年にA.Shamir博士とG.Blakley博士によって独立に提唱された概念

11人の科学者がある極秘プロジェクトに取り組んでいる。彼らは書類をキャビネットに閉じ込め、6人以上の科学者がいる場合にのみキャビネットを開くことができるようにしたいが、必要な錠前の数は最小で何個か？各科学者が持っていなければならないの鍵の数は最小で何個か？

Dをn個の断片D_1,..., D_nに分割し、以下のようにする。

① k個以上のD_i個について知っていれば、Dは容易に計算可能である。

② k-1以下のD_i個の知識はDを完全に未決定にする。

暗号鍵を安全に保管するのに役立つ

Programming Techniques — R. Rivest, Editor

How to Share a Secret

Adi Shamir
Massachusetts Institute of Technology

In this paper we show how to divide data D into n pieces in such a way that D is easily reconstructable from any k pieces, but even complete knowledge of $k-1$ pieces reveals absolutely no information about D. This technique enables the construction of robust key management schemes for cryptographic systems that can function securely and reliably even when misfortunes destroy half the pieces and security breaches expose all but one of the remaining pieces.

Key Words and Phrases: cryptography, key management, interpolation
CR Categories: 5.39, 5.6

1. Introduction

In [4], Liu considers the following problem:

Author's present address: A. Shamir, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

This research was supported by the Office of Naval Research under contract no. N00014-76-C-0366.

It is not hard to show that the minimal solution uses 462 locks and 252 keys per scientist. These numbers are clearly impractical, and they become exponentially worse when the number of scientists increases.

In this paper we generalize the problem to one in which the secret is some data D (e.g., the safe combination) and in which nonmechanical solutions (which manipulate this data) are also allowed. Our goal is to divide D into n pieces $D_1, \ldots, D_n$ in such a way that:

(1) knowledge of any k or more D_i pieces makes D easily computable;
(2) knowledge of any $k-1$ or fewer D_i pieces leaves D completely undetermined (in the sense that all its possible values are equally likely).

Such a scheme is called a (k, n) *threshold scheme*.

Efficient threshold schemes can be very helpful in the management of cryptographic keys. In order to protect data we can encrypt it, but in order to protect the encryption key we need a different method (further encryption changes the problem rather than solves it). The most secure key management scheme keeps the key in a single, well-guarded location (a computer, a human brain, or a safe). This scheme is highly unreliable since a single misfortune (a computer breakdown, sudden death, or sabotage) can make the information inaccessible. An obvious solution is to store multiple copies of the key at different locations, but this increases the danger of security breaches (computer penetration, betrayal, or human errors). By using a (k, n) threshold scheme with $n = 2k - 1$ we get a very robust key management scheme: We can recover the original key even when $\lfloor n/2 \rfloor = k - 1$ of the n pieces are destroyed, but our opponents cannot reconstruct the key even when security breaches expose $\lfloor n/2 \rfloor = k - 1$ of the remaining k pieces.

In other applications the tradeoff is not between secrecy and reliability, but between safety and convenience of use. Consider, for example, a company that digitally signs all its checks (see RSA [5]). If such an executive is given a copy of the company's secret signature key, the system is convenient but easy to misuse. If the cooperation of all the company's executives is necessary in order to sign each check, the system is safe but inconvenient. The standard solution requires at least three signatures per check, and it is easy to implement with a $(3, n)$ threshold scheme. Each executive is given a small magnetic card with one D_i piece, and the company's signature generating device accepts any three of them in order to generate (and later destroy) a temporary copy of the actual signature key D. The device does not contain any secret information and thus it need not be protected against inspection. An unfaithful executive must have at least two accomplices in order to forge the company's signature in this scheme.

Threshold schemes are ideally suited to applications in which a group of mutually suspicious individuals with conflicting interests must cooperate. Ideally we would like the cooperation to be based on mutual consent, but the veto power this mechanism gives to each member can paralyze the activities of the group. By properly choosing the k and n parameters we can give any sufficiently large majority the authority to take some action while giving any sufficiently large minority the power to block it.

2. A Simple (k, n) Threshold Scheme

Our scheme is based on polynomial interpolation: given k points in the 2-dimensional plane $(x_1, y_1), \ldots, (x_k, y_k)$ with distinct x_i's, there is one and only one polynomial $q(x)$ of degree $k-1$ such that $q(x_i) = y_i$ for all i. Without loss of generality, we can assume that the data D is (or can be made) a number. To divide it into pieces D_i, we pick a_1 random $k-1$ degree polynomial $q(x) = a_0 + a_1 x + \ldots + a_{k-1}x^{k-1}$ in which $a_0 = D$, and evaluate

$$D_1 = q(1), \ldots, D_i = q(i), \ldots, D_n = q(n).$$

Given any subset of k of these D_i values (together with their identifying indices), we can find the coefficients of $q(x)$ by interpolation, and then evaluate $D = q(0)$. Knowledge of just $k-1$ of these values, on the other hand, does not suffice in order to calculate D.

To make this claim more precise, we use modular arithmetic instead of real arithmetic. The set of integers modulo a prime number p forms a field in which interpolation is possible. Given an integer valued data D, we pick a prime p which is bigger than both D and n. The coefficients $a_1, \ldots, a_{k-1}$ in $q(x)$ are randomly chosen from a uniform distribution over the integers in $[0, p)$, and the values $D_1, \ldots, D_n$ are computed modulo p.

Let us now assume that $k-1$ of these n pieces are revealed to an opponent. For each candidate value D' in $[0, p)$ he can construct one and only one polynomial $q'(x)$ of degree $k-1$ such that $q'(0) = D'$ and $q'(i) = D_i$ for the $k-1$ given arguments. By construction, these p possible polynomials are equally likely, and thus there is absolutely nothing the opponent can deduce about the real value of D.

Efficient $O(n \log^2 n)$ algorithms for polynomial evaluation and interpolation are discussed in [3] and [3], but even the straightforward quadratic algorithms are fast enough for practical key management schemes. If the number D is long, it is advisable to break it into shorter blocks of bits (which are handled separately) in order to avoid multiprecision arithmetic operations. The blocks cannot be arbitrarily short, since the smallest usable value of p is $n+1$ (there must be at least $n+1$ distinct arguments in $[0, p)$ to evaluate $q(x)$ at). However, this is not a severe limitation since sixteen bit modulus (which can be handled by a cheap sixteen bit arithmetic unit) suffices for applications with up to 64,000 D_i pieces.

Some of the useful properties of this (k, n) threshold scheme (when compared to the mechanical locks and keys solutions) are:

(1) The size of each piece does not exceed the size of the original data.
(2) When k is kept fixed, D_i pieces can be dynamically added or deleted (e.g., when executives join or leave the company) without affecting the other D_i pieces. (A piece is deleted only when a leaving executive makes it completely inaccessible, even to himself.)
(3) It is easy to change the D_i pieces without changing the original data D—all we need is a new polynomial $q(x)$ with the same free term. A frequent change of this type can greatly enhance security since the pieces exposed by security breaches cannot be accumulated unless all of them are values of the same edition of the $q(x)$ polynomial.
(4) By using tuples of polynomial values as D_i pieces, we can get a hierarchical scheme in which the number of pieces needed to determine D depends on their importance. For example, if we give the company's president three values of $q(x)$, each vice-president two values of $q(x)$, and each executive one value of $q(x)$, then a $(1, n)$ threshold scheme enables checks to be signed either by any three executives, or by any two executives one of whom is a vice-president, or by the president alone.

A different (and somewhat less efficient) threshold scheme was recently developed by G.R. Blakley [2].

Received April 1979; revised September 1979

References
1. Aho, A., Hopcroft, J., and Ullman, J. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, Reading, Mass., 1974.
2. Blakley, G.R. Safeguarding cryptographic keys. Proc. AFIPS 1979 NCC, Vol. 48, Arlington, Va., June 1979, pp. 313–317.
3. Knuth, D. *The Art of Computer Programming*, Vol. 2. *Seminumerical Algorithms*. Addison-Wesley, Reading, Mass., 1968.
4. Liu, C.L. *Introduction to Combinatorial Mathematics*. McGraw-Hill, New York, 1968.
5. Rivest, R., Shamir, A., and Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. Comm. ACM 21, 2 (Feb. 1978), 120–126.

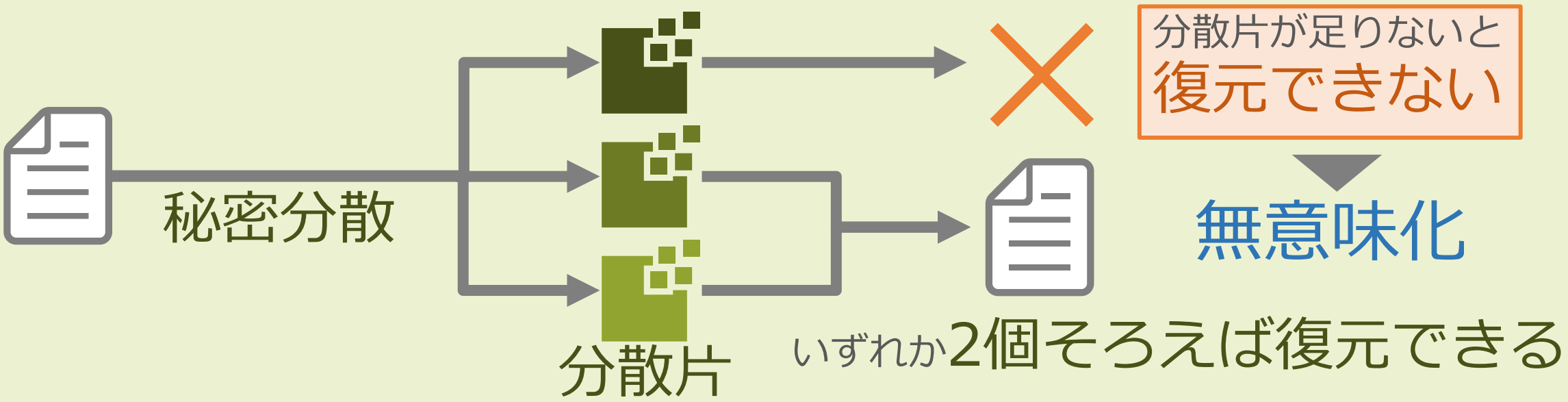
引用　https://web.mit.edu/6.857/OldStuff/Fall03/ref/Shamir-HowToShareASecret.pdf

(k, n)しきい値分散法

n 個の分散片の内 k 個そろわなければ
元の情報を復元できない符号化手法

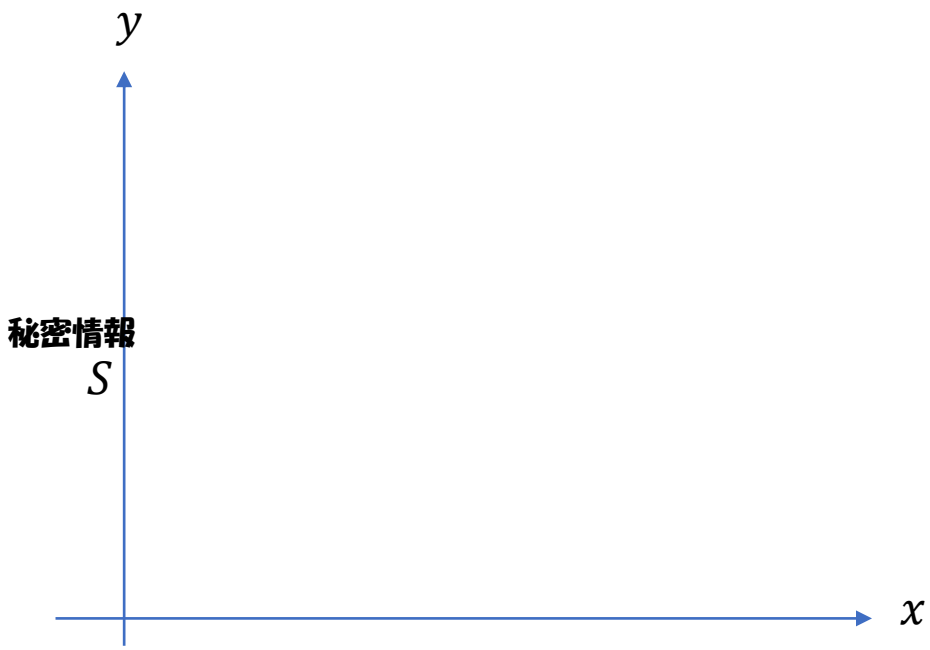
暗号化と異なり
復号のための
鍵が不要

例: 3個に分散した内2個そろわなければ元の情報を復元できない



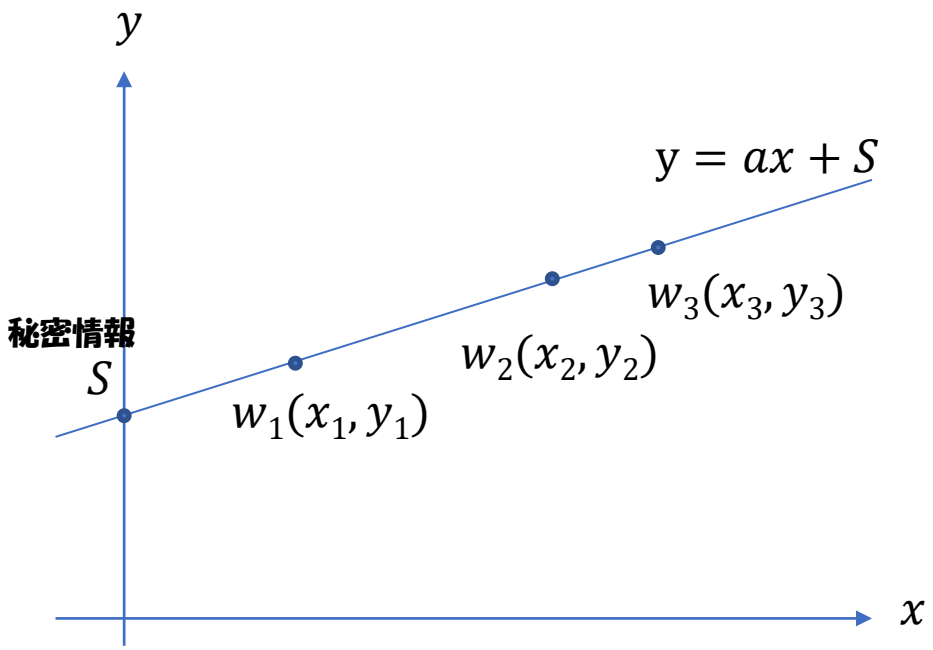
情報の分散

秘密情報Sをしきい値分散で分割(2つのシェアが集まれば復元できる)する方法(多項式を使う方法)



情報の分散

秘密情報Sをしきい値分散で分割(2つのシェアが集まれば復元できる)する方法(多項式を使う方法)



y=ax+S(aは乱数で決定)の式を生成

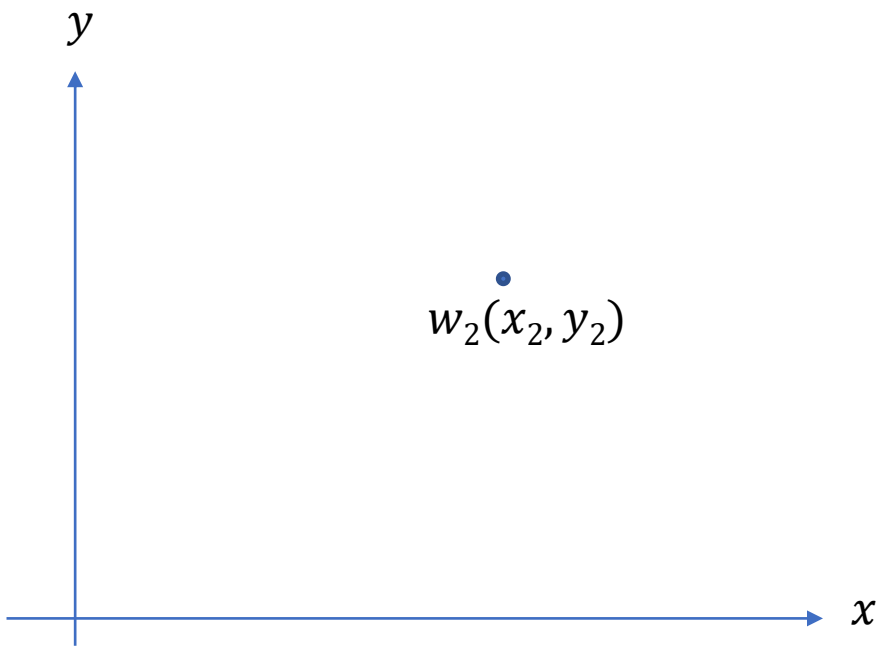
乱数でx_1,x_2,x_3の値を生成
これらの値をy=ax+Sの式に当てはめてy_1,y_2,y_3を計算

$y_1=ax_1+S$
$y_2=ax_2+S$
$y_3=ax_3+S$

これらの値を使って、w_1,w_2,w_3を定義

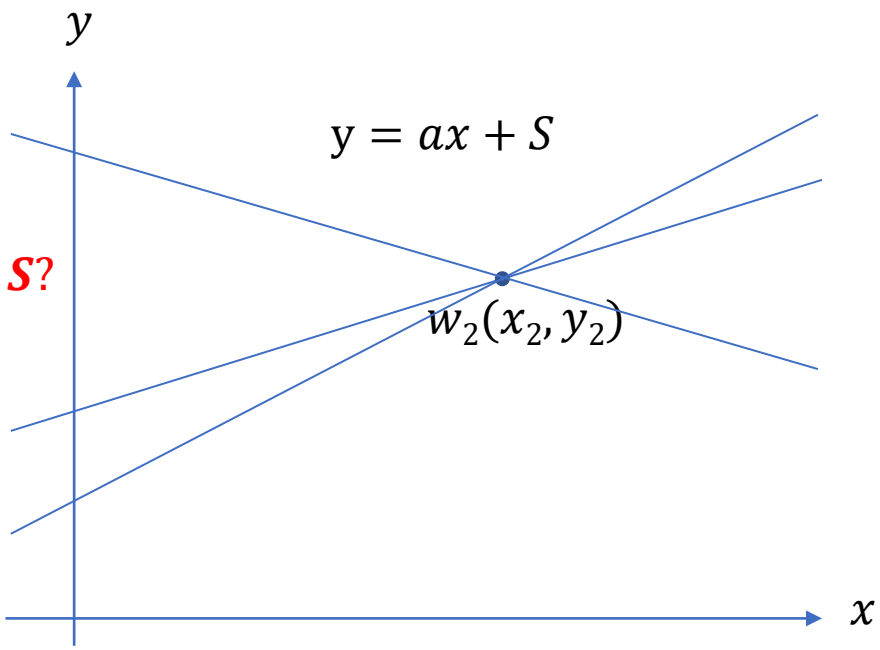
情報の復元

シェアが1個しか集まらなかったとき



情報の復元

シェアが1個しか集まらなかったとき

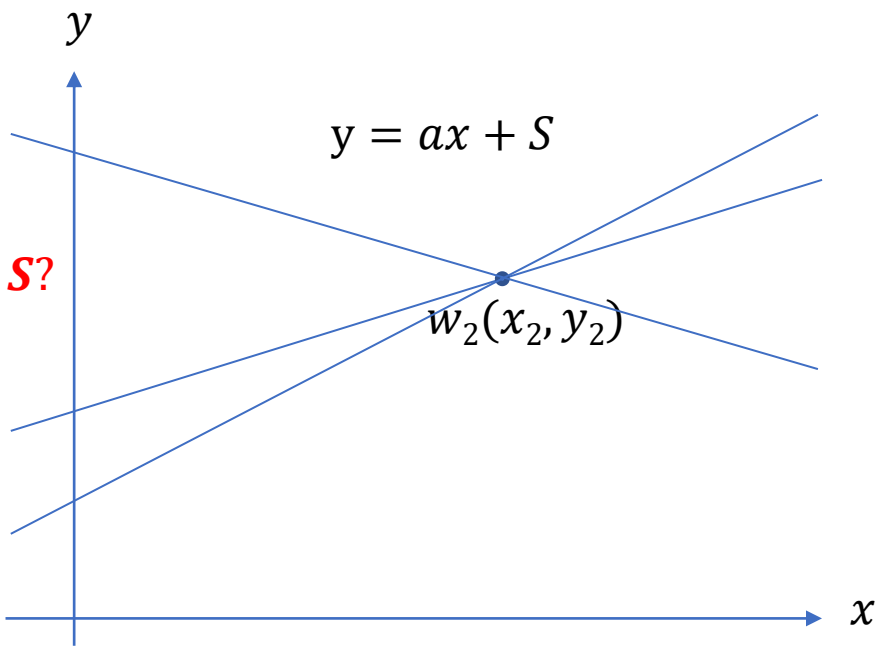


W_2を通る直線は無限に引けるから、Sがどこなのか定まらない

秘密情報Sを特定することができない

情報の復元

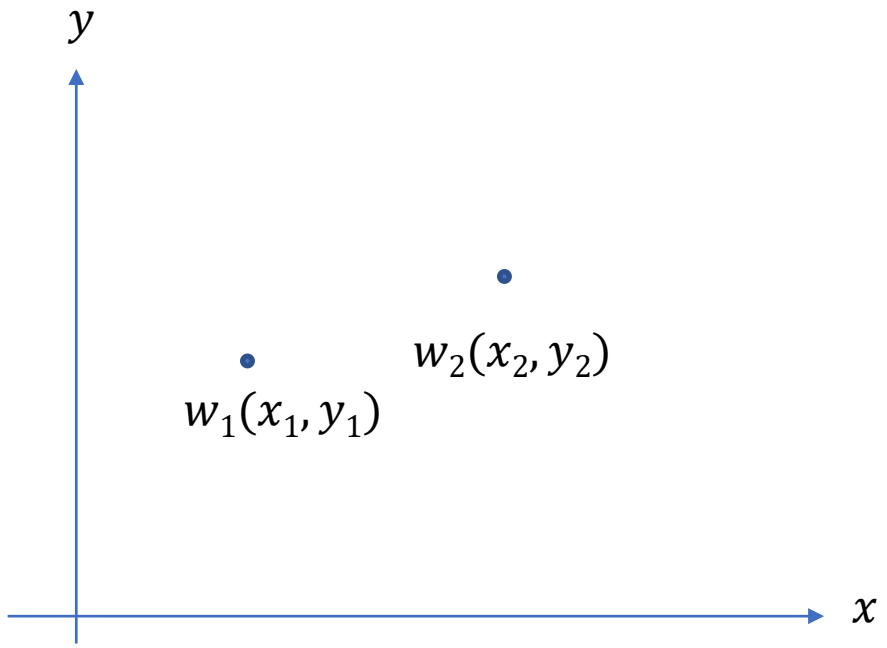
シェアが1個しか集まらなかったとき



W_2を通る直線は無限に引けるから、Sがどこなのか定まらない

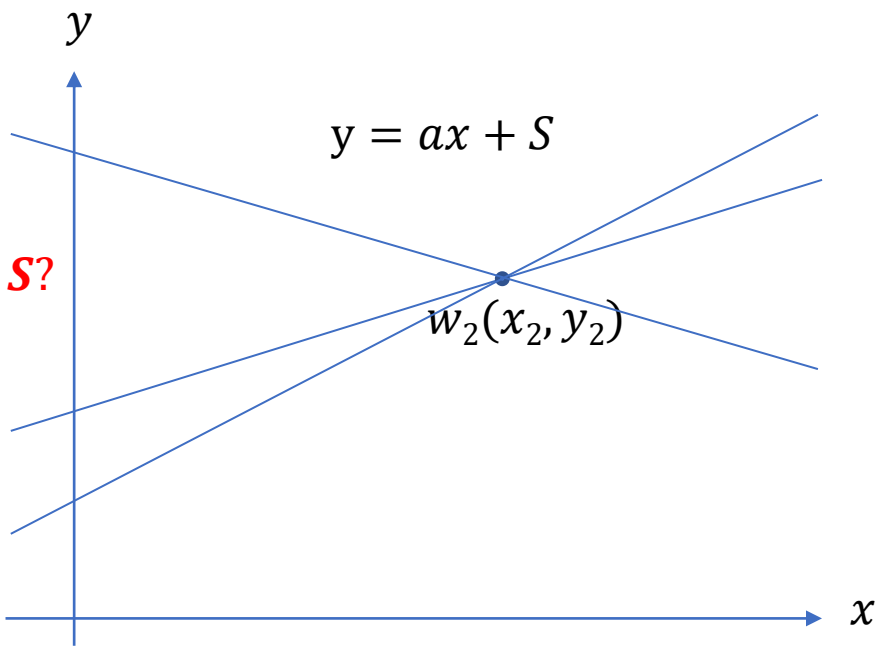
秘密情報Sを特定することができない

シェアが2個集まったとき



情報の復元

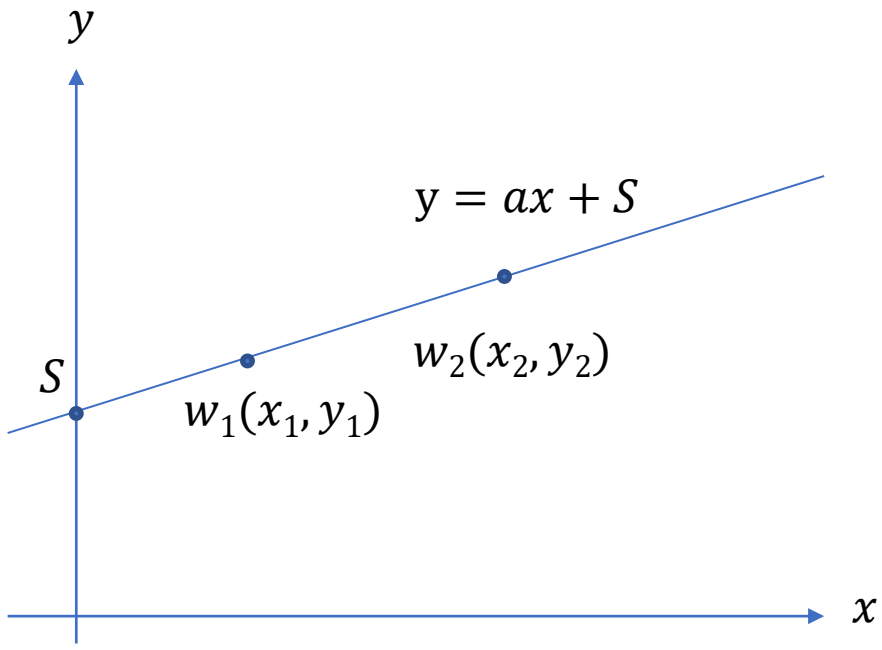
シェアが1個しか集まらなかったとき



W_2を通る直線は無限に引けるから、Sがどこなのか定まらない

秘密情報Sを特定することができない

シェアが2個集まったとき

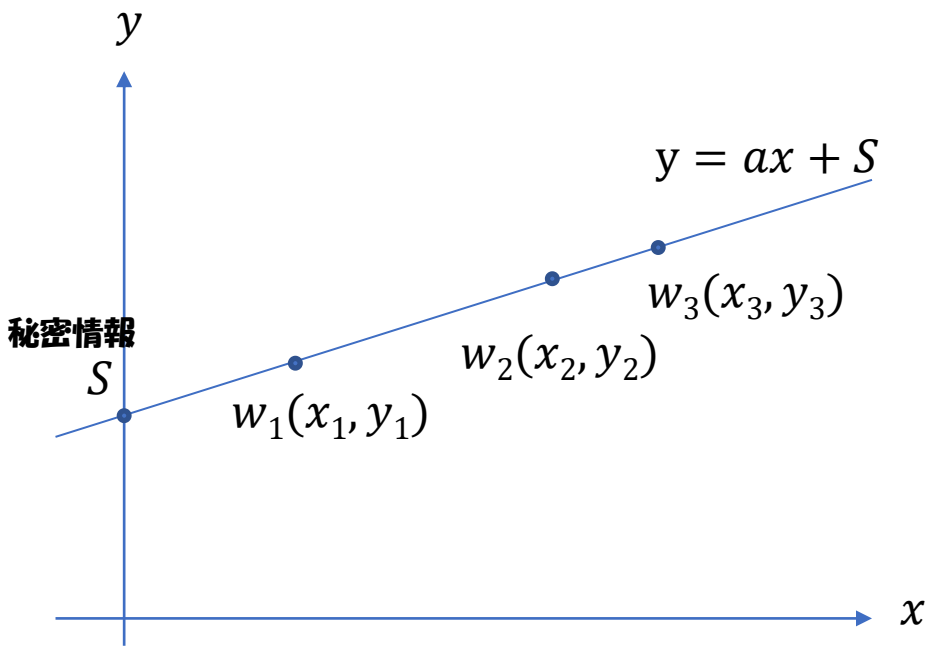


W_1、W_2を通る直線は1本だけ引けるから、Sが求められる。

秘密情報Sを復元することができる

情報の分散

秘密情報Sをしきい値分散で分割(2つのシェアが集まれば復元できる)する方法(多項式を使う方法)



y=ax+S(aは乱数で決定)の式を生成

乱数でx_1,x_2,x_3の値を生成
これらの値をy=ax+Sの式に当てはめてy_1,y_2,y_3を計算

$y_1=ax_1+S$
$y_2=ax_2+S$
$y_3=ax_3+S$

これらの値を使って、w_1,w_2,w_3を定義

Appendix

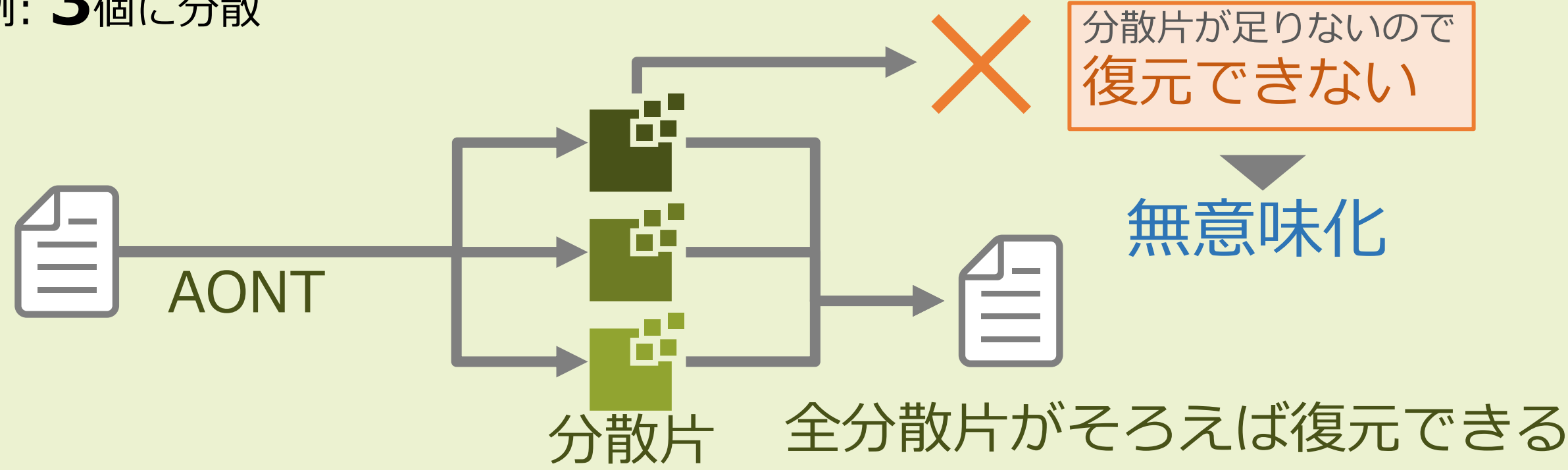
3つのシェアが集まれば復元できるように分割する場合は、$y=ax^2+bx+S$の式を使えばOK。
4つのシェアにする場合は3次式、というように、k個のシェアが集まれば復元できるようにするには、(k-1)次式を使って秘密情報Sを分割。

AONT(All Or Nothing Transform)

n 個の分散片の内全てそろわなければ
元の情報を復元できない符号化手法

暗号化と異なり
復号のための
鍵が不要

例: 3個に分散



AONT概要

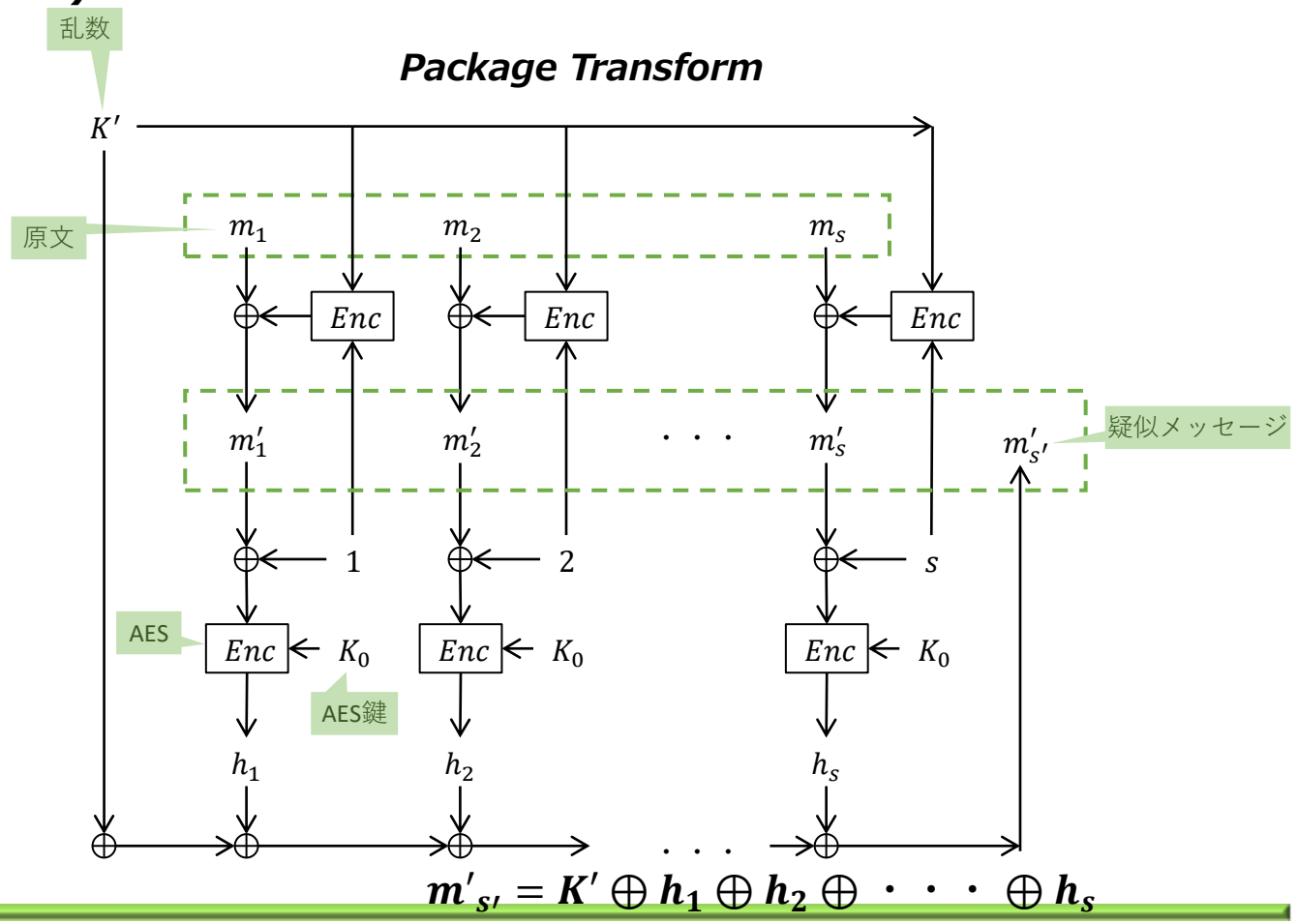
RSA暗号考案者の一人であるRivest博士によって1997年に提唱された概念

攻撃者によるbrute-force attack(総当たりによる攻撃)をより困難にするための暗号化の前処理として考えだされた

- ✓ 変換fは可逆であり、擬似メッセージ列が与えられれば、元のメッセージ列を得ることができる。
- ✓ 変換fとその逆はともに効率的に計算可能である(つまり、多項式時間で計算可能である)。
- ✓ 擬似メッセージブロックの1つでも未知であれば、任意のメッセージブロックの関数を計算することは計算上不可能である。

AONTの特徴

- ✓ 変換後の総データ量は元データとほぼ同じである。
- ✓ すべての分散片(シェア)を集めることで復号可能となる。
- ✓ 任意のサイズで分散可能である(但し、最小サイズはあり)。



しきい値分散/AONT比較

	しきい値分散	AONT
データ量	× 元データのn倍	○ 元データとほぼ同じ
難読性	◎ 情報理論的安全性	○ 計算量的安全性
可用性	○ 指定個の分散片が集まれば復元可能	× 一つでも分散片がなくなると復元不能
速度	△ 多項式方式は遅いため、XOR方式が考案されている	○ 複雑な計算は行わないため、速い

暗号化との比較

ZENMU TECH

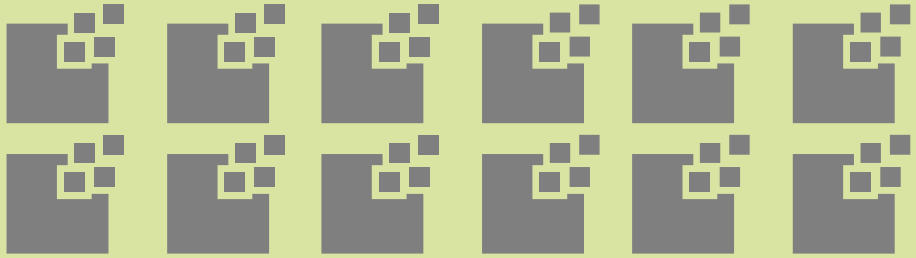
情報

暗号化

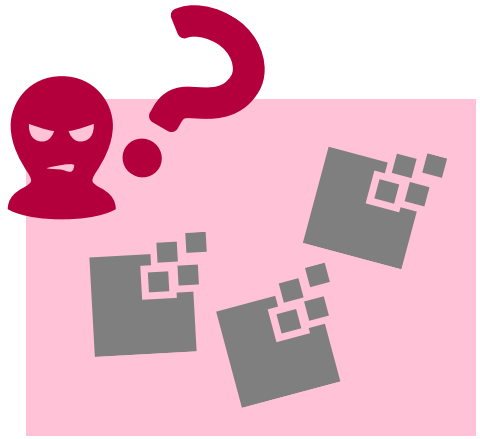
攻撃者に鍵を盗まれると解読の危険

鍵にリスクが集中

秘密分散



分散片をしきい値個そろえなければ解読不能



リスクを分散

秘密分散の国プロへの展開

SIP 戦略的イノベーション創造プログラム
Cross-ministerial Strategic Innovation Promotion Program

内閣府 Cabinet Office

戦略的イノベーション創造プログラム（SIP）第2期
課題「光・量子を活用したSociety5.0実現化技術」
採択課題、研究責任者及び研究開発機関について

（1）レーザー加工
①ＣＰＳ型レーザー加工機システム研究開発
採択課題：ＣＰＳ型レーザー加工機システムによるスマート製造推進拠点
研究責任者：小林　洋平（国立大学法人東京大学 物性研究所教授）
代表研究開発機関：国立大学法人東京大学
共同研究開発機関：（なし）

②空間光制御技術に係る研究開発
採択課題：高精度・高スループットレーザー加工のための空間光制御技術開発
研究責任者：豊田　晴義（浜松ホトニクス株式会社 中央研究所 研究主幹）
代表研究開発機関：浜松ホトニクス株式会社
共同研究開発機関：国立大学法人宇都宮大学

③フォトニック結晶レーザーに係る研究開発
採択課題：フォトニック結晶レーザーの高輝度化およびスマート化の研究開発
研究責任者：野田　進（国立大学法人京都大学 大学院工学研究科 教授）
代表研究開発機関：国立大学法人京都大学
共同研究開発機関：三菱電機株式会社、ローム株式会社

（2）光・量子通信
①量子暗号技術
採択課題：量子暗号技術と量子セキュアクラウド技術に関する研究開発
研究責任者：藤原　幹生（国立研究開発法人情報通信研究機構 未来ICT 研究所量子ICT 先端開発センター 研究マネージャー）
代表研究開発機関：国立研究開発法人情報通信研究機構
共同研究開発機関：日本電気株式会社、株式会社東芝、早稲田大学、
国立大学法人東京大学、国立大学法人北海道大学、
株式会社ZenmuTech
（以上）

06. 光・量子を活用したSociety 5.0 実現化技術

西田 直人（にしだ なおと）
(株)東芝 特別嘱託

目指す姿

概要

Society 5.0 実現には、**サイバー空間とフィジカル空間を高度に融合させるサイバーフィジカルシステム**(CPS)**の構築が鍵**。現在、IoT/AIからスマート製造へと投資が開始されているが、**社会・産業界共通の投資を阻むボトルネックが存在・我が国が強みを有す光・量子技術を活用**し、これらの**ボトルネックを解消可能な加工、情報処理、通信の重要技術を順次・開発**を行い、「レーザー加工市場シェア奪還のための日本発コア技術等の製品化」「ものづくり設計・生産工程の最適化」「高秘匿クラウドサービスの開始」等を達成し、**Society 5.0実現を加速度的に推進**させる。

目標

・**CPS型レーザー加工機システムの実現**（レーザー加工条件の初期選定のリードタイムを現在の9割減）や、**高精度・高スループットな加工を実現する空間光制御技術を実用化**（現在の10~100倍程度高速化）し、製造業における加工の世界トップの生産性を実現する。
・**フォトニック結晶レーザーの高輝度化及び高性能化を実現**し、将来のレーザー加工等への応用を見据えつつ、人や障害物をいち早く検知し安心・安全な移動を可能にする**センシング技術に活用可能な超小型光源を実現**（センシングシステムのコストを現在の9割減に貢献。現在の一般的な既存半導体レーザーの10倍の輝度を目指す）する。
・**市場競争力の高い量子暗号装置**（耐タンパ性向上、従来比4分の1の低コスト化）**を開発し**100km圏ネットワーク上で秘密分散ストレージ技術と統合することにより、完全秘匿なデータ伝送、バックアップ保管、2次利用など**新たな秘匿アプリケーションを提供する量子セキュアクラウドシステムを実現**する。
・スマート製造の実現に係る組合せ最適化等の問題の解決を世界で最も高速に実現する光電子情報処理のソフトウェアを世界に先駆けて開発する。

出口戦略

・**拠点を設立し、国内外の企業ネットワークやテストプラットフォームの提供**、技術データの収集、各企業と実装に向けた議論等を実施、**企業の評価例・採用実例等を研究開発にフィードバックして企業の事業化に結実させる**（レーザー加工）。
・機密性の高いデータを扱う医療分野やスマート製造分野のユーザと共同で試験運用し、**標準化を進め運用ガイドラインを策定する**（光・量子通信）。
・開発したソフトウェアを企業等に提供・フィードバックをもらい開発に活用するとともに、個別企業の具体的な要求に基づくソフトウェア開発につなげる（今後、詳細を決定、光電子情報処理）。
・**研究成果の積極的・戦略的な広報を実施**し、企業等に限らず社会全般へ向けて成果の浸透を図る。

社会経済インパクト

上記目標の達成を通じて、下記のような社会経済インパクトを実現する。

・日本発コア技術等の製品化による**レーザー加工市場シェアの奪還**
・ものづくり**設計・生産工程の最適化によるスマート製造の実現**
・**高機密情報の安全な流通・保管・利活用による、医療・製造分野の生産性向上**

達成に向けて

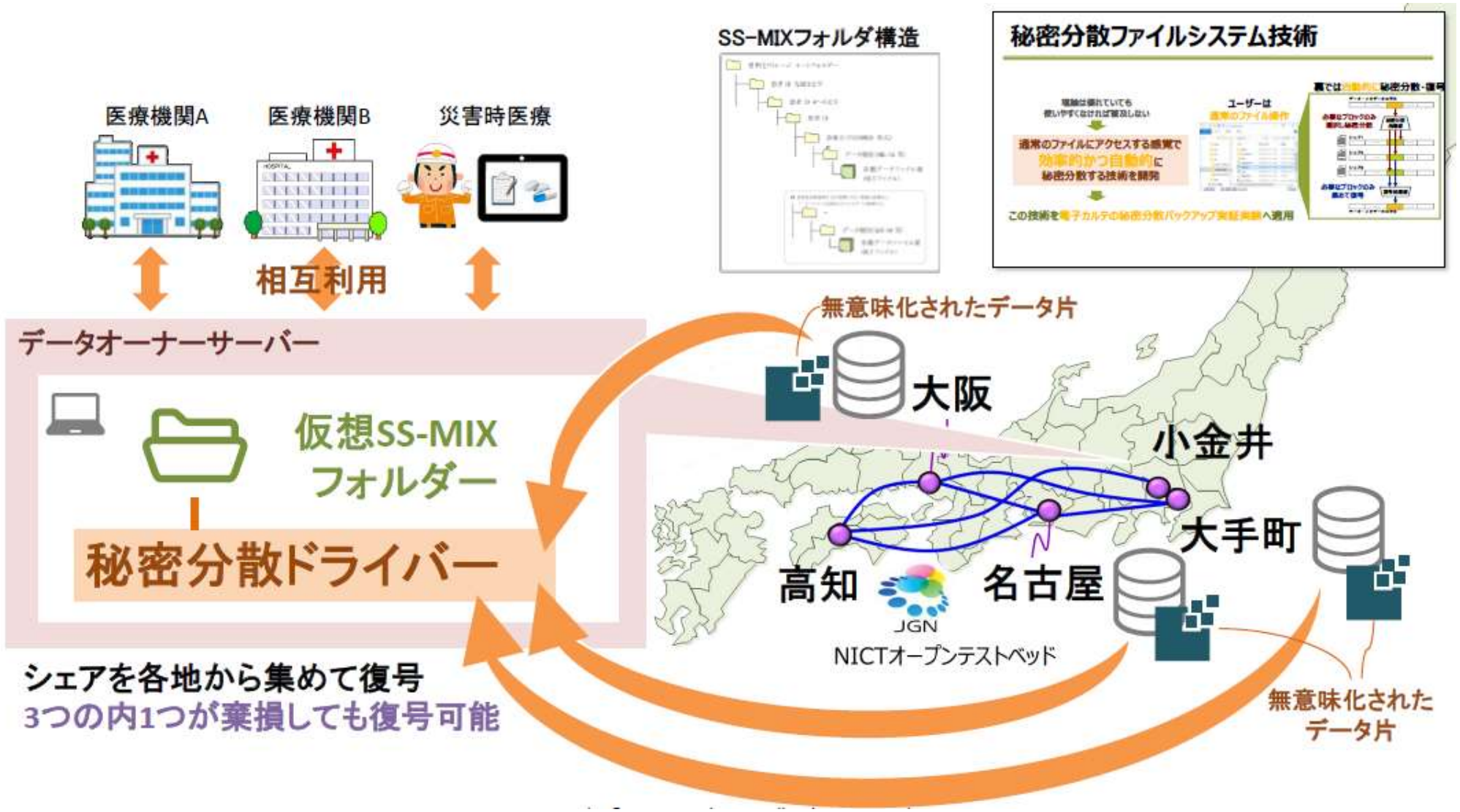
研究開発内容

Ⅰ．レーザー加工
①**サイバー（シミュレータ）とフィジカル（レーザー加工）の高度な融合**によるスマート生産の実現（特定用途のCPS（サイバーフィジカルシステム）型レーザー加工機システムの開発）
②**日本が有するコア技術「空間光制御技術」の開発**によるスマート生産の実現（高品位・高精度空間光制御技術の開発）
③日本発**フォトニック結晶レーザーの高出力化の実現**

Ⅱ．光・量子通信
○量子暗号、秘密分散、秘密計算の統合により、**解読技術の進展によるセキュリティの危殆化の懸念がないクラウドサービスの世界に先駆けた開発。**電子カルテやゲノム解析情報、スマート製造情報などを用いた**実証。**（量子セキュアクラウド技術の開発）

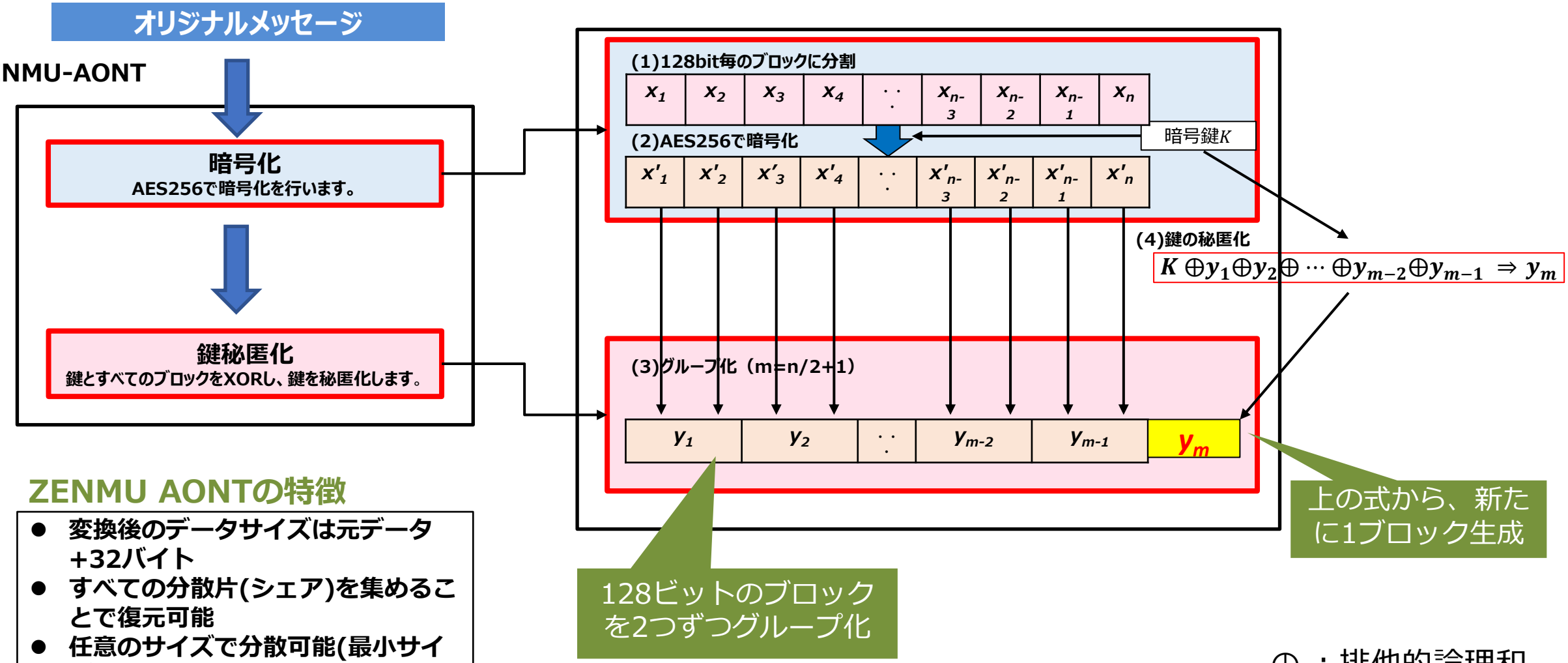
Ⅲ．光電子情報処理
○スマート製造の実現に必要な、ネットワーク上のリソースの組合せ最適化等の問題を高速で処理する光電子情報処理のソフトウェア、ミドルウェア開発を行い、クラウドサービスを実施。
※ImPACT、Q-LEAP、NEDO プロ等の状況を踏まえ、今後詳細を決定

ZENMU TECH



- ZENMU-AONT

ZENMU AONT(概要)



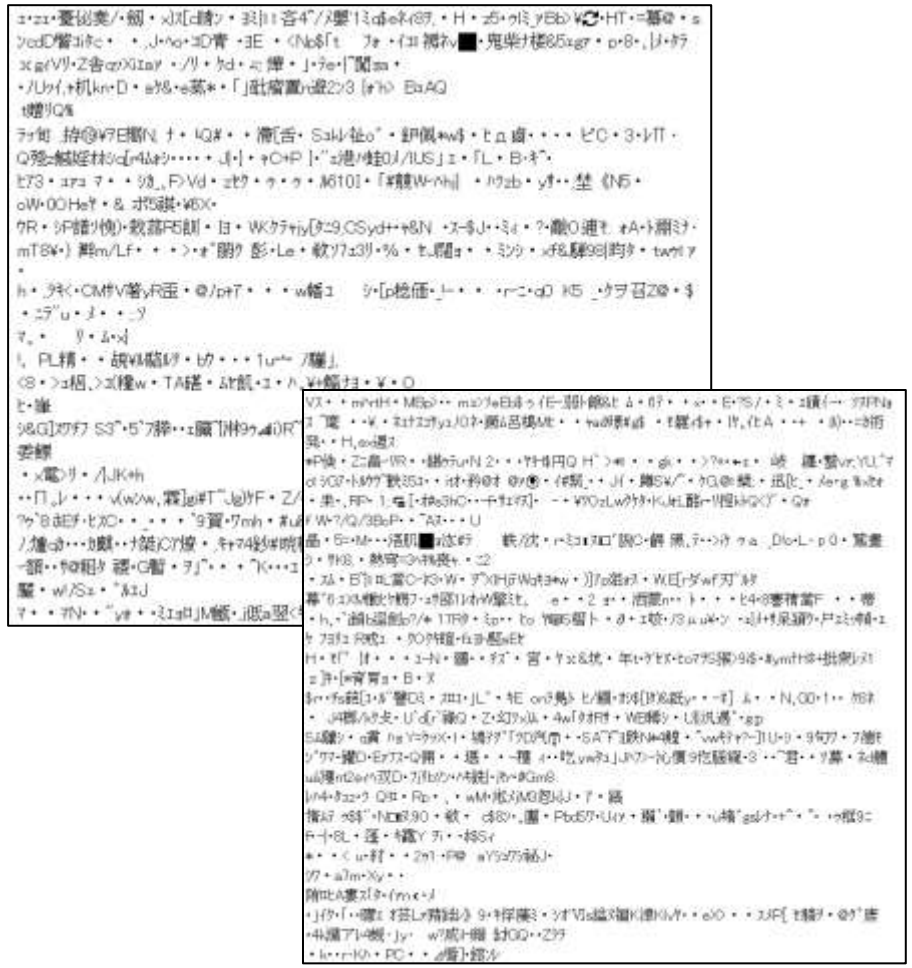
ZENMU AONTの特徴

- 変換後のデータサイズは元データ+32バイト
- すべての分散片(シェア)を集めることで復元可能
- 任意のサイズで分散可能(最小サイズは32バイト)

⊕ ：排他的論理和

ZENMU-AONT

- AONTの特徴
  - 分散割合を任意に設定可能
    - 1:10:100など
  - 一片の大きさを格段に小さくできる
    - 最小　32バイト
  - 分散後の総データ量は元データとほぼ同等
    - しきい値分散では、分散数に応じて倍々に膨らむ
  - 処理速度が速い（データ量が小さいため）



ZENMU-AONT

- AONTの特徴
  - 分散割合を任意に設定可能
    - 1:10:100など
  - 一片の大きさを格段に小さくできる
    - 最小　32バイト
  - 分散後の総データ量は元データとほぼ同等
    - しきい値分散では、分散数に応じて倍々に膨らむ
  - 処理速度が速い（データ量が小さいため）
- 安全性
  - 産業技術総合研究所　　花岡　悟一郎先生監修の自己評価書
  - WISA2019（国際学会）で、最優秀論文賞を受賞

ZENMUソリューション

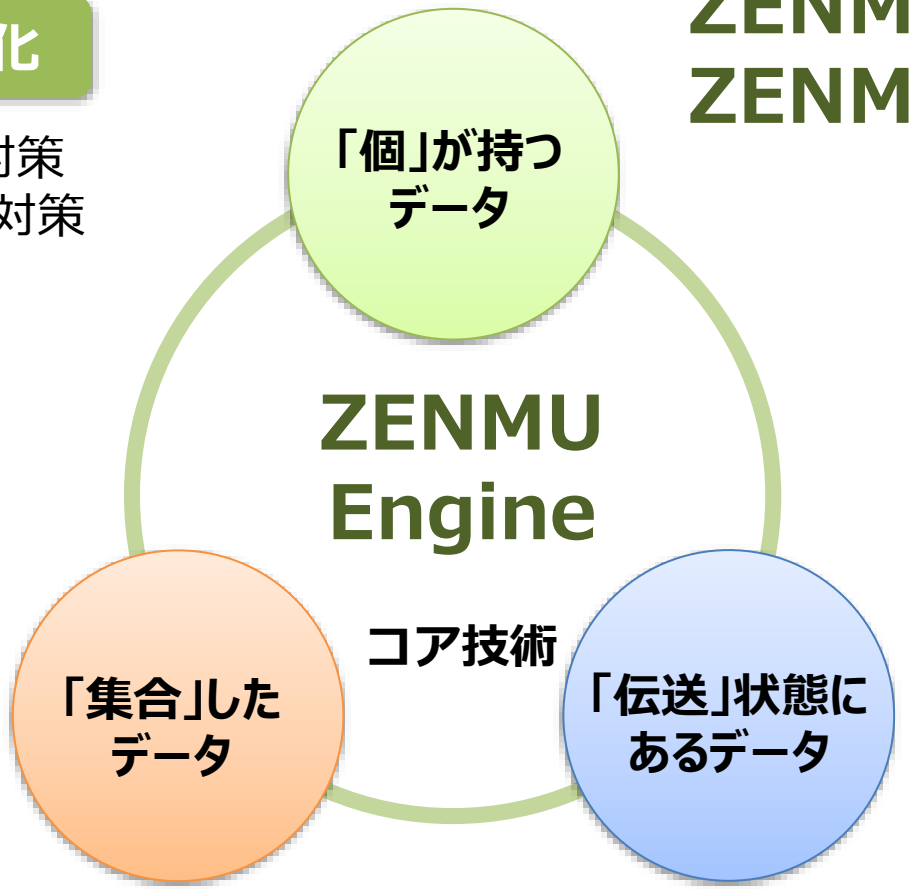
ZENMU Engine

コア技術

ZENMUソリューション

自分から離れたデバイスの無効化

・個人が持つ端末の紛失・盗難等の対策
・個人が持つ端末内のデータの漏えい対策



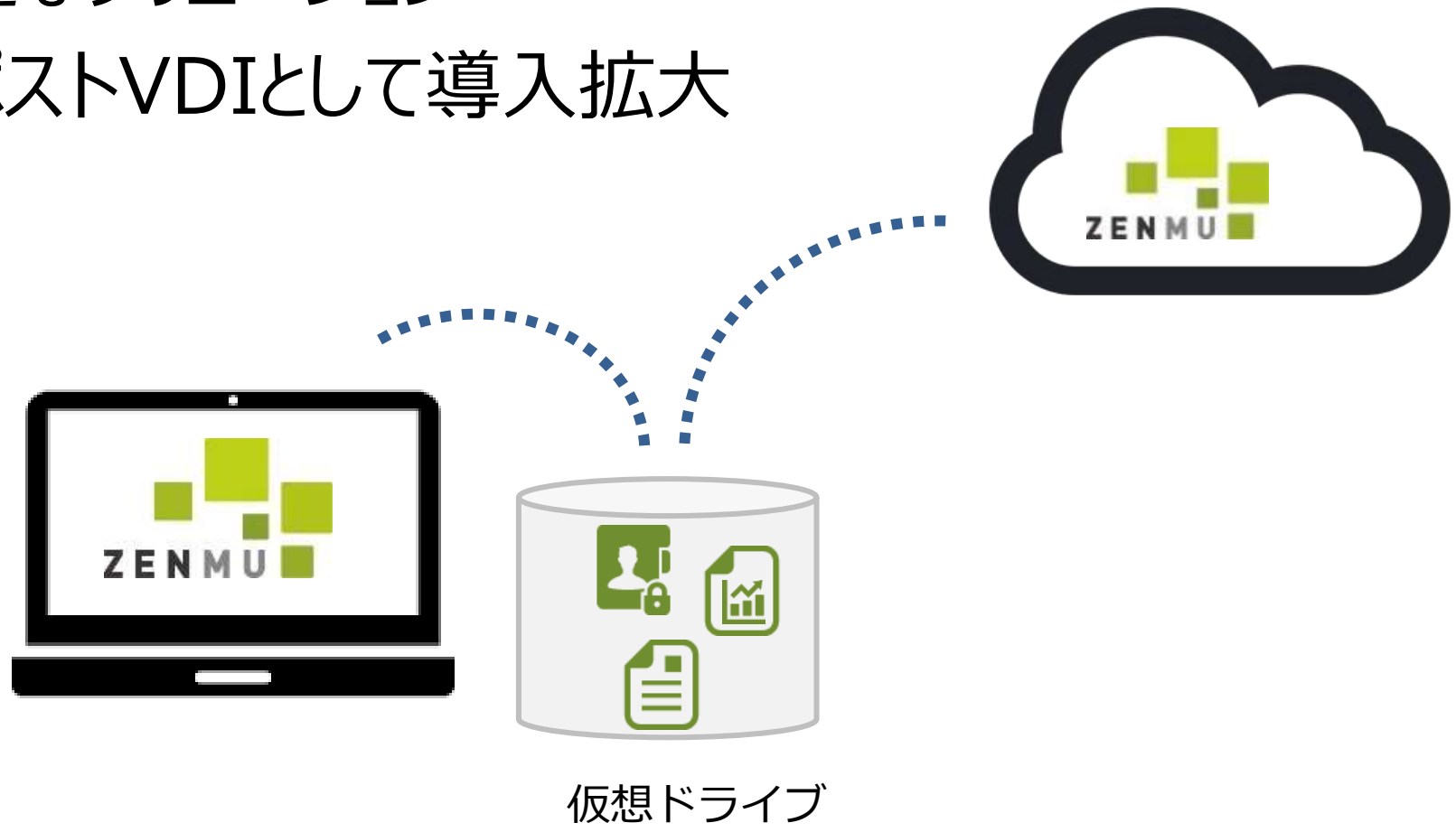
ZENMU Virtual Drive
ZENMU for PC

ZENMU Virtual Drive

- セキュアFATに最適なソリューション
- テレワーク対策やポストVDIとして導入拡大

秘密分散技術

PC上に仮想ドライブ



ZENMU Virtual Drive

ユーザー

- PC内のユーザーデータを仮想ドライブに
- 仮想ドライブを秘密分散で無意味化
- 分散ファイルをPC内とクラウド上に保管

管理者

- クラウド上でユーザー管理

クラウド接続時

クラウド上の分散ファイルと合わせてデータを復元

クラウド未接続時

未接続時、PC内には無意味な分散ファイルのみ

ZENMU Virtual Drive

オフラインでの利用

- クラウド上の分散ファイルをスマートフォンや USBメモリー、Windows共有フォルダーに同期
- ネットワーク環境によらず、オフラインでも 安定して利用可能

盗難や紛失時のロック

- 紛失に気付いた時点で本人や管理者が PCを利用停止可能
- アクセスログにより不正利用の有無を確認可能
- PCが発見された時点でロックを解除すれば すぐにPC利用再開可能

オフライン利用時

分散ファイルを スマホなどにも保管可能

管理画面

仮想ドライブロック時

分散ファイルに アクセス停止

ZENMUソリューション

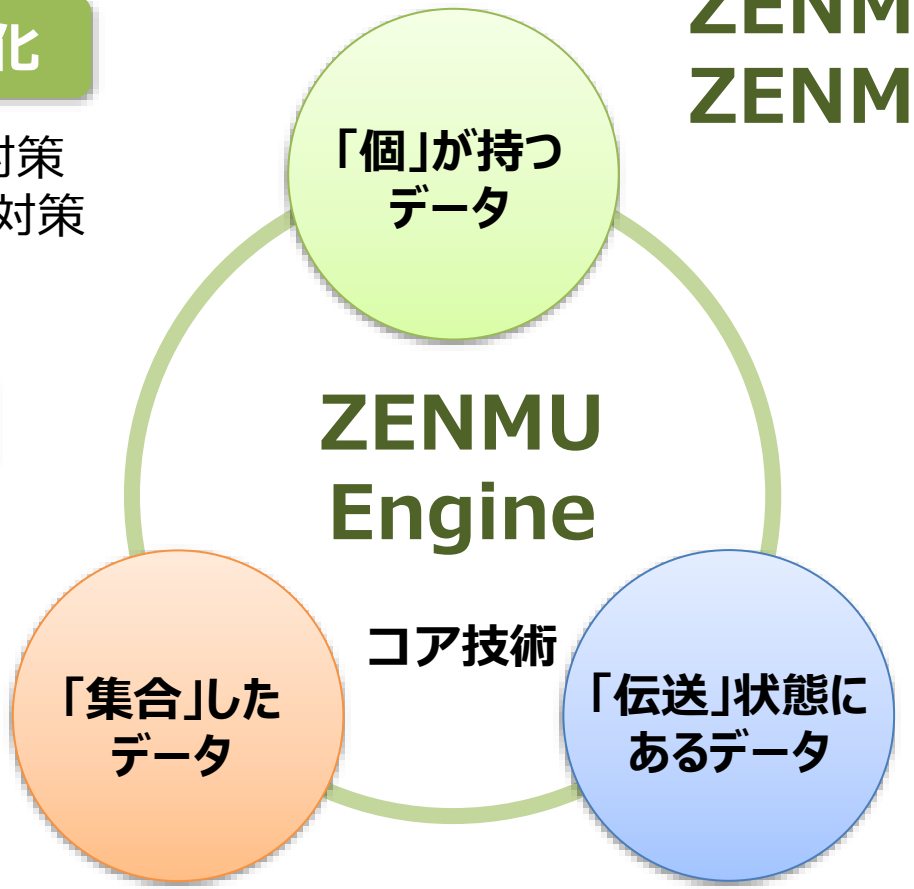
自分から離れたデバイスの無効化

・個人が持つ端末の紛失・盗難等の対策
・個人が持つ端末内のデータの漏えい対策

秘匿化によるデータの利活用

・サーバーからデータの不正搾取防止
・共同作業における本人認証など

ZENMU for Meister



ZENMU Virtual Drive
ZENMU for PC

ZENMUソリューション

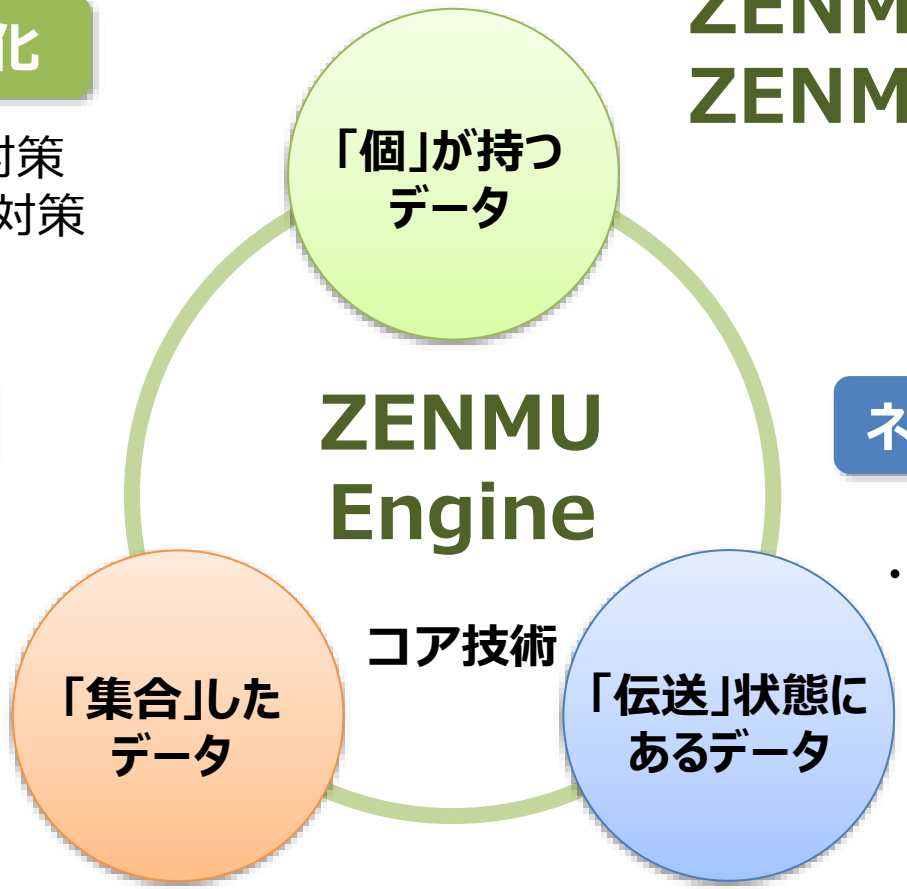
自分から離れたデバイスの無効化

・個人が持つ端末の紛失・盗難等の対策
・個人が持つ端末内のデータの漏えい対策

秘匿化によるデータの利活用

・サーバーからデータの不正搾取防止
・共同作業における本人認証など

ZENMU for Meister



ZENMU Virtual Drive
ZENMU for PC

ネットワーク上のデータの無意味化

・伝送中におけるデータの搾取/漏えい防止
例. Mail, Network通信

ZENMU for Delivery

- ZENMU Engine

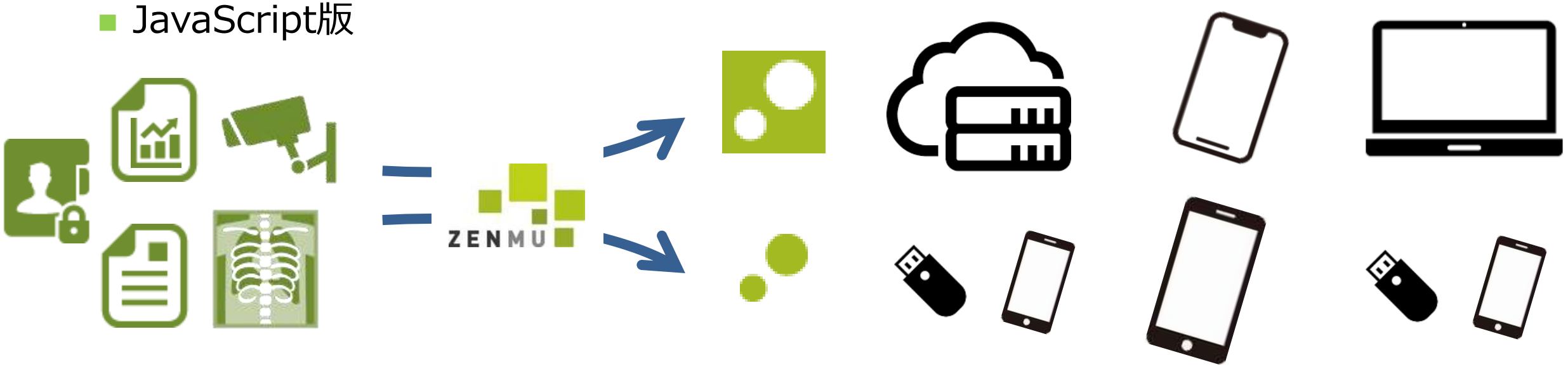
ZENMU Engine ラインナップ

ZENMUの秘密分散技術をSDKとして提供

- **ファイル単位**の秘密分散
  - ファイルやストリームデータの秘密分散処理
  - 任意の割合で複数の分散ファイルに分割・復号
- **ドライブ単位**の秘密分散
  - ZENMU for PC/ZENMU Virtual Driveに使用
  - PC内のドライブイメージを2つに分割し、SSD/HDD上と外部に保持
- **秘密分散ストレージ**
  - サーバー上のドライブを秘密分散し、ネットワーク上のストレージに分散保管
  - 利用者からは、ファイルシステムとしてアクセス可能

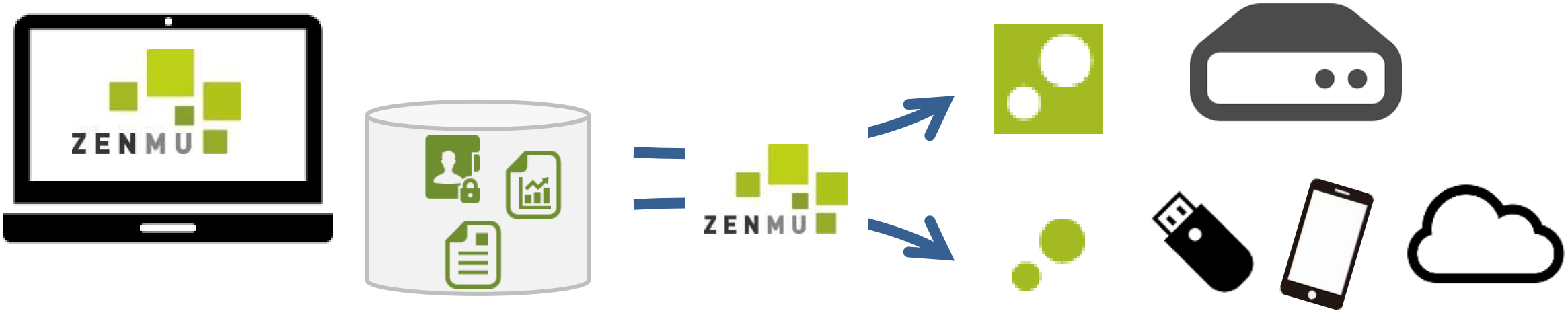
ファイル単位の秘密分散（ZENMU Engine）

- 主な機能
  - ファイルの秘密分散処理を行い、任意の割合で複数の分散ファイルに保管する機能
  - 複数の分散ファイルから復号化処理を行い、元のファイルに戻す機能
- 対応プラットフォーム
  - Windows/Linux 版
  - Java版
  - JavaScript版



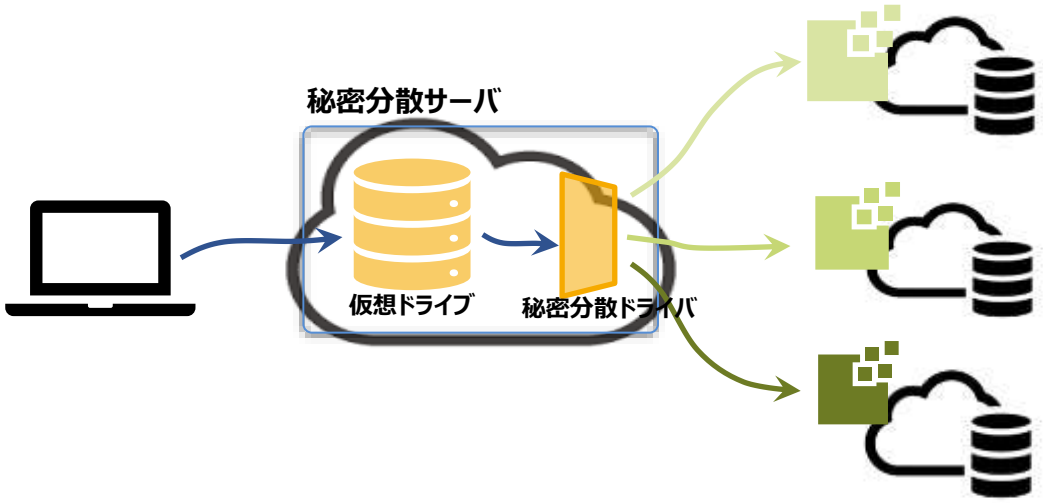
ドライブ単位の秘密分散（ZenmuDisk）

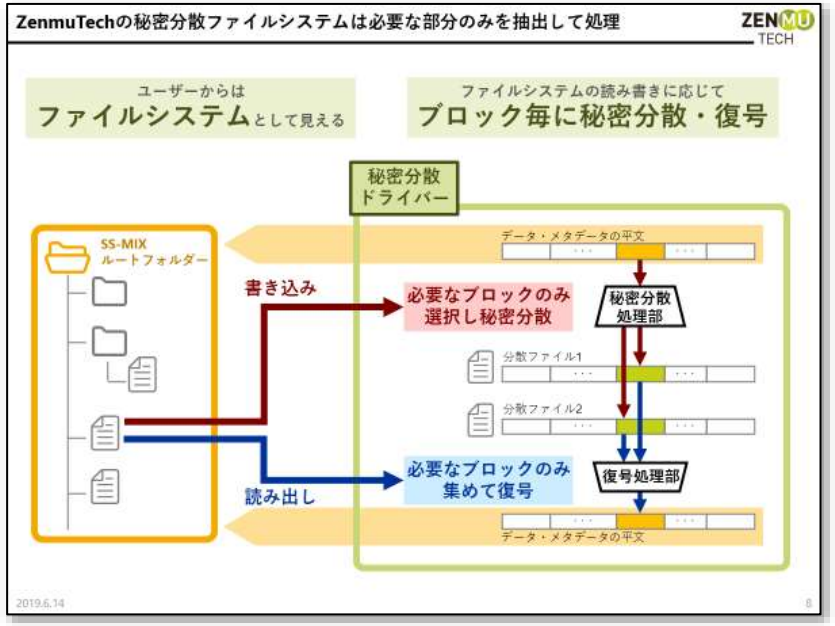
- 主な機能
  - 仮想ドライブのイメージを2つに分割し、分散ファイル1をSSD/HDD上に、分散ファイル2をメモリー上に保持
  - 仮想ドライブを表示し利用可能（マウント）/非表示し利用不可（アンマウント）にする
  - 分散ファイルの不整合を解消するためのロールバック機能。最後にアンマウントした時点または最後にコミットした時点どちらか新しい状態に戻す



秘密分散ストレージシステム

- 主な機能
  - 秘密分散サーバ上に構築された仮想ドライブを秘密分散し、ネットワーク上のストレージにシェア(分散片)を保管
  - 仮想ドライブの必要な部分のみを秘密分散処理するため、高速処理が可能
  - 利用者は、通常のファイルサーバーアクセス





- ZENMU Engine 利用例

ZENMU Engineパートナーソリューション

FUJITSU

PCデータの保護 「Local Data Protection」

NRI

デジタルウォレット開発キット 「Walletech」

HITACHI
Inspire the Next

「グローバルセキュア転送サービス」

「秘密分散 フォー メール」

「秘密分散 フォー ビデオ」

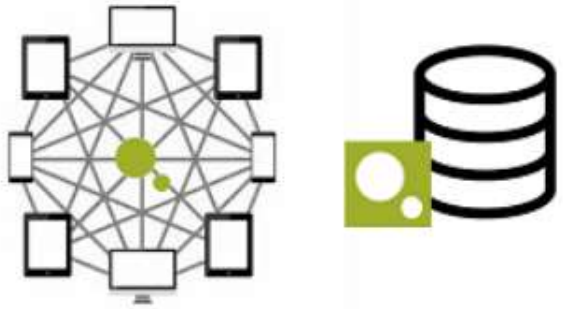
NextWare　iROBOTICS

完全データセキュリティ 「インテグリティ・ドローン」

ZENMU Engineの適用分野

1 家や車のキーをウェアラブルデバイスに分散

2 ブロックチェーンとの連携



3 IoT機器のデータ収集での通信コストの削減

4 安心・安全なパブリッククラウドの利用

セキュリティだけではなく、エンターテイメント分野などでも

有難うございました

https://zenmutech.com