ストレージネットワーキング・インダストリ・
アソシエーション日本支部

テクニカル・ホワイトペーパー

ストレージ初学者のための
ホワイトペーパー

使用にあたって

SNIA 日本支部では、本書の使用を個人に対しては個人的利用に限定して許可し、法人およびその他の事業主体に対しては社内利用（社内での複製、配布、および掲示を含む）に限定して許可する。ただし、次の要件が満たされていることを前提とする。

1. テキスト、図、チャート、表、または定義を複製する場合は、変更を加えずに全体を複製すること
2. 本書からの資料（または本書の一部）を複製した印刷文書または電子文書は、その資料に対する SNIA 日本支部の著作権を表示し、SNIA 日本支部から再利用の許可を得ていることを明示すること

上記で明示的に規定されている場合を除き、本書の商業的利用、本書の一部または全部の販売、または本書の第三者への配布を行ってはならない。明示的に付与されていないすべての権利は、明示的に SNIA 日本支部に留保されている。上記以外の目的での本書の使用の許可は、office@snia-j.org に電子メールを送付して要請する。電子メールには、要請する個人および／または法人の識別情報と、要請する使用の目的、性質、および範囲の簡単な説明を含めること。

免責事項

この文書に含まれる情報は、事前の通知なく変更される場合がある。SNIA 日本支部はこのドキュメントに関していかなる種類の保証も行わない。これには商品性および特定の目的に対する適合性の暗黙的保証が含まれるが、これらに限定されない。SNIA 日本支部は、本書に含まれる誤りあるいはこのドキュメントの交付、履行、または使用に関連した偶発的または結果的損害

に対して責任を負わない。改訂に関する提案は、office@snia-j.org まで

目次

第１章

ストレージとは

データは、コンピュータ・スマートフォン・タブレットだけでなく各種センサーなど様々なところで生み出されています。この生み出されたデータは、そのまま利用されるだけでなく複数データと結合されたり AI などにより高度な知能を生み出す元にもなります。このような、コンピュータを使った情報処理において重要な役割を担うデータを保存しておく機器がストレージです。

このストレージには、データを失わないために、たとえ障害が発生しても安心・安全にデータを保存し続けることが求められます。これに応えるため、ストレージの内部ではデータを複製化するなどの工夫によりデータを保護しています。しかし、一方でデータは年々増え続けています。そのため、増え続けるデータを長期間保存するためのコストも下げたいとの要望もあります。さらには、大規模化するデータを如何に早く処理できるかが、コンピュータにおける情報処理の速度において非常に重要な要素となるため、高性能なアクセスも求められます。

このように、ストレージは安心・安全なデータ保護を要望されつつ、低コスト・高性能と矛盾するような要望をうけ、技術開発されてきた機器なのです。しかし、あらゆる要望を全て満たすような万能なストレージは存在しません。これらの要望に応えるため、適用環境の違いやアプローチの違いによ

り、様々なストレージが存在します。非常に信頼性の高いデータ保護と高性能なストレージや、低コストだが性能は低いストレージなどの特徴をもったストレージが多く存在します。

万能なストレージが存在しないため、私たちは各種のストレージの特徴を理解し、データの価値に応じて適材適所にストレージを選択し利用することが必要になります。例えば、株価のデータなど高速に処理をすることで利益を生み出すようなデータは、高コスト払ってでも高性能なストレージにデータを格納する方が価値は高くなります。一方で、通常ではほとんどアクセスされることがないバックアップデータや法令などで定められた長期間保存するデータなどは、低コストなストレージに保存する方が良いでしょう。

このような適材適所なストレージの使い分けを行うためにも、ストレージの技術を理解することは必要不可欠です。本ドキュメントでは、これまでストレージにあまり触れてこなかったエンジニア向けに、ストレージの基礎を解説します。ストレージの正しい知識を身につけ、上手にストレージを使いこなしましょう。

第 2 章

情報ストレージの歴史

本章では情報ストレージの歴史について解説します。

情報ストレージを広く情報を記録する仕組みと捉えると、数万年前に描かれた洞窟壁画が起源と言えます [1]。ここでは、情報がデジタル化もしくは電子化されて、記録されるようになってから以降の近代の情報ストレージの歴史について振り返ります。情報ストレージの歴史については、文献 [2]、サイト [3] にも詳しい記載があるので興味のある方は参照して下さい。

情報ストレージは、様々な技術の集合によりそのシステムが実現されていて、それらは複雑に連携しています。ここでは、ストレージデバイス、ストレージアーキテクチャに分解して解説します。

2.1　ストレージデバイスの歴史

近代のストレージデバイスは、記録する情報のデジタル化及び、読み書きの電子化が行われ、またそれらの高密度化によって技術的な発展を遂げてきました。主流のストレージデバイスの物理的特徴は、紙、磁気記録、光学記録、半導体と変遷してきました。以下、代表的なストレージデバイスについて、時系列順に解説します。

以下では「現在」という記載がありますが、本書の執筆時期である「2024

年時点」とお読み取り下さい。

2.1.1 パンチカード (18 世紀-20 世紀後半)

パンチカードは、穴を空けたカードを用いて情報を記録・保存する仕組みです。初期の電子計算機のストレージデバイスとしてよく知られているが、元々は電子計算機用に発明されたものではありません。18 世紀初頭の自動織機の制御情報を記録したカードがパンチカードの起源となっています。最も代表的な自動織機は 19 世紀初頭のジャカード織機です。この時点ではカードの情報の読み取りは機械式でした。1880 年代アメリカの統計学者ホレリスにより、パンチカードの穴を電気的に検出する仕組みが発明され、国勢調査の集計用に開発されたタビュレーティング・マシンに応用されました。さらにその後、電子計算機用のストレージデバイスとして応用されました。

2.1.2 磁気テープ (1951 年 – 現在)

磁気テープは、磁性体を塗布したテープ状のフィルムを用いて磁気ヘッドにより情報を記録・保存する仕組みです。元々は録音用に設計・開発されましたが、1950 年代には電子計算機の補助記憶装置として利用されるようになりました。1951 年の UNIVAC I 計算機向けの UNISERVO I テープドライブが、初めての商用計算機向けテープドライブです。この磁気記録技術により、電子計算機のストレージデバイスが、読み込みだけでなく書き込みも電子化されたと言えます。現在もアーカイブ、バックアップ用途を中心として利用されています。

2.1.3 ハードディスクドライブ (HDD) (1956 年 – 現在)

HDD は、磁性体を塗布したドーナツ状の金属の円盤 (プラッター) 部分と読み書きを行うドライブ部分とで構成されます。円盤を回転させながら、ド

ライブ部分のアーム上の磁気ヘッドにより読み込みと書き込みを行います。この仕組みにより、磁気テープの弱点であったランダムアクセスが出来るようになりました。1956 年に IBM から出荷された RAMAC 305 が、初めてのハードディスクドライブ製品です。岩崎俊一氏 (東北大学名誉教授) によって発明された垂直磁気記録等、多くの技術革新により大容量化が実現しました。現在では 10TB を超える容量を持つ商用製品があります。現在もオンライン用途で主流のストレージデバイスであり、サイズは 3.5 インチ、2.5 インチが一般的です。

2.1.4 フロッピーディスク (1970 年頃 – 2000 年頃)

フロッピーディスクは、金属のプラッターの代わりに磁気フィルムを用いたメディアです。メディアと読み書きを行うドライブ部分とが分離していることが特徴です。メディアの入れ替えが可能であることから、ポータブルなストレージデバイスとして、メインフレームからパーソナルコンピュータまで広く使用されました。1967 年に IBM により開発され、1972 年に 8 インチサイズのフロッピーディスクが発売されました。その後、年代と共に 5.25 インチ、3.5 インチとダウンサイジングしてきました。現在は後で述べるフラッシュメモリを活用した USB メモリやネットワークの普及に伴って、ほとんど使われていません。

2.1.5 光学ディスク (1980 年頃ー現在)

光学ディスクは、レーザを使用して情報を読み取る仕組みです。当初は読み取り専用のメディアでしたが、技術革新に伴い、一回のみ書き込み可能なメディア (-R) や、複数回書き換え可能なメディア (-RE) が開発されました。年代と共に、CD (Conpact Disc), DVD (Digital Versatile Disc), BD (Blu-ray Disc) といった仕様が策定されてきました。1 層の BD は 25GB の容量を持ちます。部分的な書き込みには適していないので、アーカイブ、

バックアップ用途を中心として利用されています。

2.1.6　フラッシュメモリ (1990 年頃ー現在)

フラッシュメモリは、半導体素子により電子的にデータを保存する仕組みです。1980 年代に東芝の舛岡富士雄氏 (東北大学名誉教授) が発明しました。1989 年にインテルが NOR 型、1991 年に東芝が NAND 型の製品を発売しました。従来のストレージデバイスと比較すると、可動部がなくなった点が大きな特徴となっています。つまり、情報の読み書きの処理がすべて電子的に行われるため、読み書きのレイテンシが大幅に改善されました。情報格納単位であるセルの高密度化、多層化、多ビット化などにより大容量化が進んでいます。このフラッシュメモリは、メモリカード、USB メモリ、後述するソリッドステートドライブ等、様々な製品にコンポーネントとして利用されています。

2.1.7　ソリッドステートドライブ (SSD) (1991 年ー現在)

SSD は、フラッシュメモリをコンポーネントとして、形状 (フォームファクタ)、I/F、プロトコルといった外部仕様が従来 HDD と互換のデバイスです。形状は 2.5 インチが一般的でしたが、近年これに加えて M.2, mSATA, U.2 といった形状がサポートされています。現在、2.5 インチタイプの SSD の最大容量は 10TB を超えています。

2.2　ストレージアーキテクチャの歴史

ストレージアーキテクチャは、前述したデバイスの進化や、その他ネットワーク等の周辺技術の革新に伴って発展を遂げてきました。以下、代表的なストレージアーキテクチャについて、時系列順に解説します。ここでは主にストレージに対してデータの入出力を行う計算機 (ホスト、クライアント) との接続形態に着目して述べます。

2.2.1 ダイレクトアタッチドストレージ (DAS) (1950 年頃ー現在)

当初ストレージは計算機の周辺機器 (サブシステム) という位置づけでした。計算機とストレージは直接接続する形態を取っていました。つまり、ストレージはある一つの計算機に占有される形態となっていました。この接続形態を DAS と呼びます。現在でも、PC やサーバの内蔵ストレージではこの接続形態が用いられています。

2.2.2 ストレージエリアネットワーク (SAN) (1990 年頃ー現在)

ストレージアレイの大型化とネットワーク技術の進展に伴い、複数の計算機から同一のストレージデバイスにアクセスできる技術が開発されました。SCSI プロトコルをファイバーチャネル (FC) プロトコルでカプセル化することで、ネットワークを通じて複数の計算機とデータを送受信できるようになりました。この接続形態を SAN と呼びます。当初はストレージ専用の FC ネットワークでしたが、現在は IP ネットワークにも拡張されています。現在もミッションクリティカルな用途を中心にデータセンターで用いられています。

2.2.3 ネットワークアタッチドストレージ (NAS) (2000 年頃ー現在)

複数の計算機に対して IP ネットワークを介してファイルサービスを行うストレージを NAS と呼びます。NAS がサポートする代表的なファイルプロトコルは NFS, CIFS/SMB です。NFS は Linux, UNIX 系で、CIFS/SMB は Windows 系でよく用いられています。SAN と異なり専用のネットワークを用意する必要がないことから導入コストを抑えることができ、データセ

ンターに加えてコンシューマ用途でも用いられることが多いです。

2.2.4 ユニファイドストレージ (2002 年頃ー現在)

ストレージシステムの低コスト化をさらに実現するために、SAN と NAS を 1 つに統合した形態のシステムをユニファイドストレージと呼びます。具体的なコストメリットとしては、SAN 用途 (ブロック) と NAS 用途 (ファイル) との間での容量などリソースの分割損が解消される、用途ごとでのストレージシステムの導入を不要化できる、管理するストレージシステムの数を減らせるなどがあります。

2.2.5 オブジェクトストレージ (2006 年頃ー現在)

これまで述べてきたアーキテクチャは、主に計算機やデータセンターの内部で用いられています。インターネット技術の発展に伴い、インターネット経由でのストレージサービスの需要が高まってきました。これに応えるのが http プロトコルを用いてデータアクセスを行うオブジェクトストレージです。インターネット黎明期には ftp を用いたファイル共有サービスが中心でしたが、現在ではオブジェクトストレージをベースとした http によるストレージサービスが多くなっています。

第 3 章

ストレージシステム

情報化時代とインターネットの普及に伴い、オンラインコンテンツなどが増加するにつれてデータの量が増加し、より多くのストレージが必要になっています。この様な大容量データを保存するように構成されたのがストレージシステムです。

ストレージシステムは、大きくサーバ等に直接接続する方式（DAS）とネットワーク経由で接続する方式に分類されます。さらに、ネットワーク経由で接続する方式としては SAN、NAS およびオブジェクトストレージが存在します。それぞれ次のような特徴があります。

3.1 DAS（Direct Attached Storage）

DAS の特徴は、サーバとブロックストレージを専用のコントローラとケーブルを使用して接続します。DAS は安価で設置と操作が簡単であると一般的には言われています。DAS とサーバの接続にはいくつか方式があり、以下のようになります。

- SAS（Serial Attached SCSI）
- USB（Universal Serial Bus）

- FC-AL（Fibre Channel Arbitrated loop）
- P2P（Point to Point）

DAS とサーバは 1 対 1 で接続されるため、ストレージを複数サーバで共有することはできません。そのため、ストレージと接続されたサーバが障害などで停止すると、ストレージへのアクセスも停止します。その問題を回避して、冗長性を取るために複数のケーブルを利用してストレージにアクセスするという方法を取ることができます。しかしながら、この方式ではストレージにアクセスするためのサーバなどが、ストレージ側のケーブルを接続するポート一つにつき、一つに制限されてしまうという問題があります。スマホや PC などに内蔵されているものとは異なり、このように外部接続を行い利用されるストレージ機器は基本的に高性能であるために、複数のデバイスで共有して利用したいという要望が多いのですが、この DAS 方式ではそのような要望に応えることが出来ません。

3.2 SAN（Storage Area Network）

SAN は DAS とは異なり、ブロックストレージをネットワークで接続し、ストレージを複数サーバで共有することができます。接続方法はいくつか方法があり、以下の例が挙げられます。

- FC SAN（Fibre Channel Storage Network）
- IP SAN（Internet Protocol Storage Network）

FC SAN は Fibre Channel Protocol に対応した機器を必要とし、ネットワーク機器として FC Switch、サーバのインターフェースとして FC HBA (Host Bus Adapter) が必要になります。IP SAN は一般的な Ethernet 環境で構成され、ネットワークスイッチは安価な Ethernet Switch、サーバのインターフェースは NIC（Network Interface Card）で構成できます。

これまで紹介してきた DAS と SAN はブロックアクセス方式と呼ばれま

す。その理由は、サーバなどが DAS や SAN を利用してデータにアクセスする際に、そのアクセス単位となっているものがドライブ上のデータブロックであるためです。つまり、データのやり取りはある特定の領域（ブロック）単位で行われており、その領域（ブロック）に何が書かれているのか、もしくは何を書き込むのかということはストレージは全く意識しておらず、命令に従ってデータの読み書きを淡々と行うタイプのものです。

3.3 NAS（Network Attached Storage）

NAS は世間での認知度は比較的高いと考えられます。家電量販店でも NAS という名前で販売されているものが多々ありますし、自宅で NAS を使ってファイル保存や、テレビ録画などを実施している方もいると思います。

NAS は DAS や SAN とは異なり、ファイルストレージをネットワークで接続し、複数のサーバでファイルを共有する事ができます。主な利用用途は、ファイルサーバや Web コンテンツの格納などです。NAS は、大量のトランザクションが発生する環境や、大量のファイルが一つのディレクトリに配置される環境において、性能が著しく低下してしまう可能性があります。

3.4 オブジェクトストレージ

データの保存をオブジェクト単位で行い、一般的にそのデータには IP ネットワークを介してアクセスを行うストレージです。NAS にイメージは近いのですが、データの保存方式がファイルではなく、オブジェクトと呼ばれる一意の ID を持ったデータである点が大きく異なります。ファイルアクセス方式に必要なファイルシステム相当を比較的シンプルで柔軟に設計することが可能であり、NAS のようにファイル単位で管理するストレージよりも数多くのデータを格納することが出来ます。そのために、特に動画や写真などサイズが大きく、数も増えるデータに適しているストレージであり、近年増加している大規模なデータの格納場所として広く利用されています。

第 4 章

ストレージのコンポーネント

ストレージは一般に、以下のコンポーネントにより構成されています。

- ネットワークポート
- コントローラ
- キャッシュ
- ドライブ

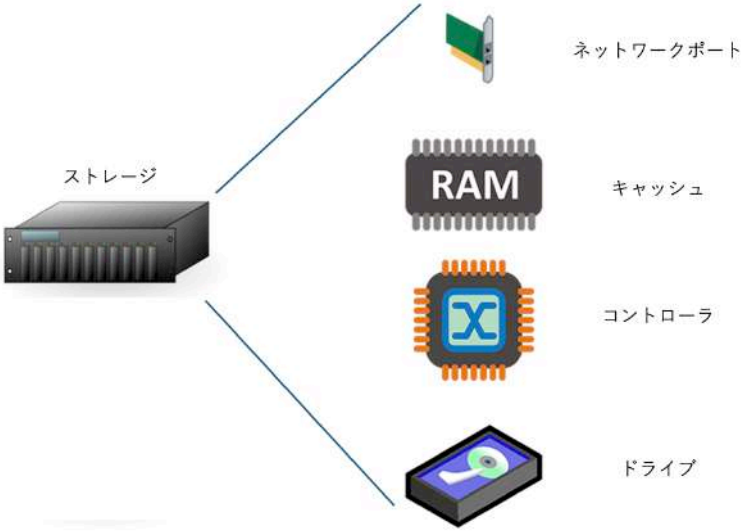


図 4.1: ストレージのコンポーネント

4.1 ネットワークポート

ネットワークポートは、サーバや他のストレージと通信を行う接続口です。

サーバがストレージのデータにアクセスする際には、このネットワークポートに対してアクセス要求を送信し、以降このポートを通してデータの書き込み・読み取り処理が行われます。

またストレージのネットワークポート同士を接続し、相互にデータのバックアップを行う機能を提供する製品もあります。

そのほか、管理や操作を行うためのインタフェースとなるのもネットワークポートの役割です。運用管理者は専用の Web インタフェースやアプリケーションを通してこのネットワークポートにアクセスします。

導入時には利用環境や業務要件に合わせた最適な設定を行い、また運用中には使用状況の確認や、機器故障時の管理者やストレージベンダへの通知がこのネットワークポートを通して行われます。

多くのストレージでは複数のネットワークポートを冗長化させることで、１つのポートが故障しても業務や管理が継続できるようになっています。

4.2 コントローラ

コントローラは「制御部」という名前が示す通り、全体を制御しストレージとしての機能を提供するコンポーネントです。

コントローラの最も大きな役割は、ネットワークポートを通して受信したデータをドライブに書き込み、またはドライブからの読み取り処理を行い、その結果をサーバに応答することです。

コントローラでは大きくは以下のような処理が行われています。

4.2.1 ストレージ・プロトコル対応

サーバとストレージが正しく通信するには、相互に同じ（互換性のある）プロトコル・通信手続きを使う必要があります。コントローラはネットワークポートを通してサーバから受信した指示を解釈して処理を行い、規定に則ってその結果をサーバに応答します。

対応可能なインタフェースは製品特性や企図する用途によって異なります。要件や用途に従って適切なインタフェースに対応したプロトコルを選択する必要があります。

4.2.2 データ保護

ストレージでは受信したデータを複数ドライブに分散して書き込みを行うことで、性能や部品故障への耐久性を向上させています。コントローラは受信したデータに対して万一の破損に備え復元のためのデータを付加し、分割や並べ替えなどを行って分散書き込みに適した形に整形処理を行います。

データ保護の詳細については後述の章をご参照ください。

4.2.3 付加価値の提供

ストレージには、性能・容量だけでなくバックアップ機能や各種 OS・仮想サーバ (VM)・コンテナへの対応、また近年ではコンプライアンスや格納コストの削減、SDGs への配慮といった複雑な課題への対応が求められています。こういった要件に対応すべく、コントローラでは以下のような機能を提供するための処理も行われています。

- バックアップ（スナップショット、レプリケーションなど）
- 暗号化
- データ圧縮・重複排除（重複データの削減）

* 各種 OS・仮想サーバ (VM)・コンテナ向けプラグイン、API への対応など

4.3 キャッシュ

キャッシュはコントローラとドライブの間で一時的にデータを貯める領域で、ストレージ製品の性能を向上させるうえで重要な役割を果たします。通常、SSD や DRAM など、高速なメディアやメモリが使われます。

キャッシュでは主に以下のようなことが行われています。

- よく使われるデータを格納し、ドライブへのアクセスを減らす
- データの圧縮や暗号化など、最終保管前のデータ加工を行う
- データをまとめて整形し、ドライブへの書き込み効率を上げる

上述の通り、キャッシュにはドライブに書き込まれる前のデータが保管されています。

データがドライブに書き込まれる前に停電などで失われることのないように、キャッシュにはバッテリーが付いていたり、専用のバックアップ領域を持つ NVRAM とよばれる特殊なメモリが使われていたりします。

4.4 ドライブ

ドライブは、ストレージにおける最終的なデータ保管先です。

回転する磁気ディスクに磁気ヘッドでデータを書き込むハードディスクや、フラッシュメモリにデータを書き込む SSD（ソリッド・ステート・ドライブ）などが使われています。

容量当たりの単価ではハードディスクの方が安く、一方で容量当たりの性能や電力消費量では SSD が優れています。ただし SSD については技術進化により容量単価の改善が進み、差はかなり少なくなってきています。個人用 PC もいまではすっかり SSD が標準ですね。

SSD に使われているフラッシュメモリには論理上書き込み回数に制限があることから、かつては主業務向けストレージには耐久度の高い（=高価な）SSD でないと不安という認識が一般的でした。しかしその普及にともなってストレージベンダでの使いこなし技術の進化や利用の実態把握も進み、今では SSD タイプの選定基準は性能や容量単価が重視される傾向があります。

ドライブの選定にあたっては近年では、設置面積あたりの容量や、発熱

量、あるいは消費電力あたりの性能といった環境面の指標も一般的になりつつあります。

性能や容量、価格が要件を満たすことはもちろん大前提ですが、上記のような観点も踏まえて選定するとより良い選択になるでしょう。

第 5 章

ストレージ管理

ストレージを運用していくなかで、ストレージに対して様々な管理が必要になってきます。以下に例を挙げます。

- 容量管理
- 性能管理
- 障害管理
- バージョン管理

5.1 容量管理

ストレージに保管するデータ量を事前に算出できる場合は、それを元にストレージ全体の容量を設計し運用することができますが、ストレージを使っていくうちに設計した容量を超えてしまう事は多々あります。そこで、ストレージの容量を管理する必要があります。

5.2 性能管理

ストレージの性能を管理します。利用するストレージ製品によって取得できる性能情報にばらつきがありますが、以下の情報を取得することにより、ストレージの限界を早く知る事ができます。

- IOPS
- Latency(msec)
- スループット (MB/s 等)

また、ストレージの本格稼働前に事前に時間的な余裕がある場合は、ストレージの負荷試験を実施し、ストレージの限界性能を知っておくと指標を作るのが安易になります。

5.3 障害管理

可用性が高いと言われるストレージでも、部品の故障による障害は発生します。例えば HDD や SSD、ストレージコントローラで使われている CPU・メモリ・各種インターフェースカードなどが挙げられ，さらに光ケーブルで接続する場合はそのトランシーバなども挙げられます。これらの故障にすぐに気づき、速やかに対処することがストレージ運用者に求められます。障害管理の方法はいくつかありますが、ストレージ製品のベンダー保守を利用する場合と、自分たちでストレージ障害を監視するための仕組みを作る方法があります。

5.4 バージョン管理

ストレージ装置の OS(Operating System) やファームウェアには、不具合が含まれていることがあります。不具合の内容は軽微な問題や致命的な問

題があります。これらの不具合を未然に防ぎ、安定したストレージ運用を実現するためには、ストレージで使われている OS やファームウェアのバージョンを管理します。具体的な方法は、ストレージ製品ベンダーの保守サービスを利用する他に、自分たちでストレージ製品の不具合情報を参照し不具合を認知する方法があります。そこで見つけた不具合を吟味し、必要ならば迅速にバージョンアップの計画を立て、その計画を実行し、重大な障害を未然に回避する必要があります。

第 6 章

データ保護

ストレージではデータを保護する必要があります。どのようなことが起きようともデータが無くならないようにしなくてはなりません。

その実現方法に関してはバックアップやレプリケーションの実施、ディザスタリカバリプラン（災害復旧プラン）の作成など多々ありますが、この章ではもう少し基本的な部分、ドライブやサーバー/ノードレベルでの保護についてみていきます。

6.1 RAID（Redundant Arrays of Inexpensive Disks）

RAID という概念は「A Case for Redundant Arrays of Inexpensive Disks (RAID)」（日本語に訳すと「冗長化された安価なディスクの配列」）という 20 世紀後半の論文で発表されました。この論文のタイトルから分かるように、RAID は、複数のディスクを統合することで単一のディスクよりも高い性能や信頼性を提供するという設計思想に基づいています。

論文の中では 1 から 5 までの 5 つの RAID レベルが提案されましたが、現実的な実装では RAID2 と RAID4 は採用されていません。また、論文中にはないのですが、後に RAID0 と RAID6 が加わり、これらを含む RAID0、

1、3、5、6 が広く利用されています（とはいえこの中で RAID3 は利用されることが少なくなってきています [理由は後述]）。

RAID0

ストライピングとも呼ばれ、複数のディスクを一つのディスクとみなして処理を行い、大容量を実現する手法です。注意して頂きたいのは、RAID と言いながらもこの方式ではデータの冗長性は提供されません。

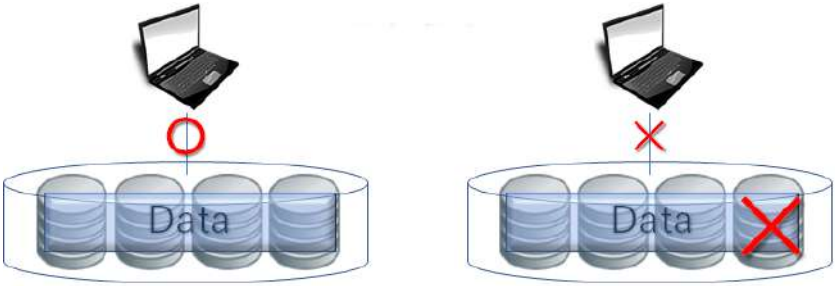


図 6.1: RAID0

RAID1

ミラーリングと呼ばれ、同じデータを 2 本のディスクに同時に保存することで冗長性を確保します。これにより、単一のディスク障害が発生してもデータの安全性は保たれます。また、比較的シンプルな構成であるためにデータアクセスの高速化を見込むことが出来ますが、必要なディスク容量が書き込むデータ量の約 2 倍必要になるというデメリットもあります。

図 6.2: RAID1

RAID3 と RAID5

パリティ情報を使用して 1 本のディスク故障までデータ保護ができる冗長性を確保する手法です。パリティ情報を保存するために追加で 1 本分のディスクを必要とします（例：ディスク 4 本分の容量が必要な場合は 4+1 の合計 5 本のディスクを準備する必要があります）。RAID3 はビットまたはバイトベースで、RAID5 はより細かいブロックベースでデータの冗長性を確保します。ブロックベースでの処理が多い現代のコンピュータ環境では、RAID5 が一般的に好まれます。

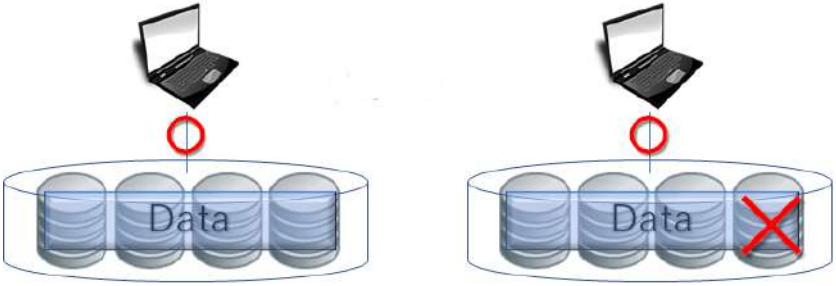


図 6.3: RAID3 と RAID5

RAID6

RAID6 は、2 本のディスクが同時に故障してもデータを保護できる高度な冗長性を提供します。これは RAID の中では新しい技術であり、前述の通り初期の RAID の論文では考慮されていませんでした。

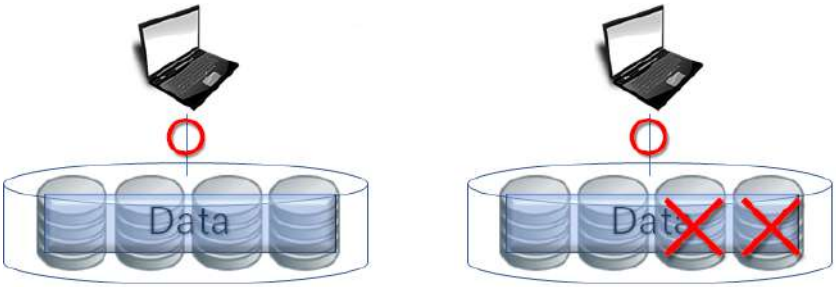


図 6.4: RAID6

■コラム: パリティ計算

パリティ情報によって何故データが保護出来るのかについて、簡単な例を紹介します。これによって何故パリティ情報があるとデータを保護/復元できるかのイメージを持ってもらえると嬉しいです。

例として 5 本からなる 4+1 の RAID5 ドライブ（群）があるとします。また、データはデジタル情報として持っており、今回は正解データが 0 か 1 かだけが判明すれば良いこととします。

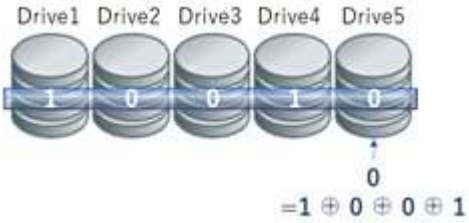
また、パリティによるデータ保護に利用される、xor 計算の定義は以下の通りです。

「xor (⊕) 計算は 1 もしくは 0 のどちらかの値を取り、計算対象が同じものである場合には 0、異なる場合には 1 の値を取る。すなわち以下の関係が成り立つ」

$1 \oplus 1 = 0$
$1 \oplus 0 = 1$
$0 \oplus 1 = 1$
$0 \oplus 0 = 0$

今当該 RAID5 ドライブ（群）のデータ保存用ドライブ 4 本に、それぞれ Drive1: 1, Drive2: 0, Drive3: 0, Drive4: 1 というように 0 か 1 の値が書き込まれたとします。

この例の場合のパリティ値は、Drive1 から Drive4 までの数字の xor 計算をすることにより求められます。すなわち
$1 \oplus 0 \oplus 0 \oplus 1 = 0$
を計算すればよいこととなり、その結果は 0 となります。つまりパリティとして 5 本目のドライブ、Drive5 には 0 という値が入り、RAID5 での保護が完成します。



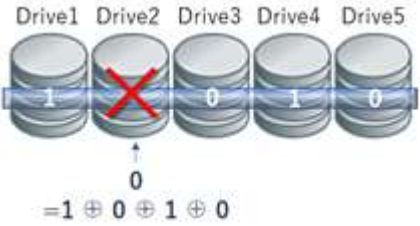
この状態で Drive2 が故障してデータがなくなったとします。

この Drive2 のデータを復活させるためには、その値が 0 だったのか 1 だったのかが分かれば良いということになるのですが、それは残ったドライブデータの xor 計算をすれば判明します。これがパリティによるデータの保護です。

実際に計算してみましょう。残った Drive1、3、4、5 データの xor 計算をすると

1 ⊕ 0 ⊕ 1 ⊕ 0= 0

となり、故障した Drive2 に入っていたデータは 0 であることが分かります。



Drive2 以外のドライブで故障が発生した例においても、実際に計算をしてみると全て問題なくデータを復帰することが出来るので、興味がある方は計算をしてみてください。パリティによるデータ保護のイメージがつかみやすくなると思います。

6.2 イレージャーコーディング（Erasure Coding: EC）

イレージャーコーディングの基礎となる技術は基本的に RAID と同様です。保存するデータにデータ復元のためのパリティ相当の情報を追加し、それら分散させて保存することによりデータの保護を目指します。

しかしながら RAID と大きく異なる点は、分散されたデータを RAID グループを組むディスク群の中以外にも保存が出来るようにしている所です。そのために、例えば同じサイトや異なるサイトにある複数ノード/サーバーに分散してデータを保存するということが可能です。

これにより、フレキシブルな構成やデータ保護の手法がとれる上に、データを複数拠点に分散することによりデータ保護を行うことが可能です。このような理由から複数拠点に分産されていることが多いクラウド系のストレージサービスなどでは積極的に利用されているテクノロジーです。

しかしながら、一つのデータを複数拠点/ノード/サーバーに分散して持っているが故に、特にそれぞれの分散データの配置間距離が大きい場合に、データの読み書きに関するパフォーマンスが低下するケースもあります。

第 7 章

バックアップ

一般にストレージにおいては、**喪失からの復旧**を目的に、**データを複製**することをバックアップと呼びます。

ストレージ製品がいかに優れていたとしても、人為的なミス、不正アクセスや災害など、様々な要因によりデータが失われるリスクをゼロにすることはできません。

データの重要度や日常の使い方、想定する喪失リスクに基づいて、どのようなバックアップを行うか決定します。

ここでは以下の順にバックアップの検討にあたって考慮すべき基本的な事項を説明します。

- 要件の検討：リカバリーシナリオ、想定する事象、ベストプラクティス
- 方式の決定：バックアップ方式の検討

まず要件を決定し、次いでそれを満たすための方式を検討します。

7.1 要件 1 リカバリー要件の検討：RPO、RTO、RLO

最適なバックアップ方法の選定にあたっては、まずはリカバリー（復旧）のための要件を明確にすることが重要です。

リカバリーのための要件として最も一般的な考え方は、RPO、RTO、RLO です。

RPO(Recovery Point Objective)

いつの時点のデータに戻すか？

RTO(Recovery Time Objective)

何分・何時間以内に業務を再開させるか？

RLO(Recovery Level Objective)

業務はどのくらいの復旧レベルで再開させるか？

このうち、特に RPO と RTO がストレージに強く依存します。

RPO を小さくすることは、失われるデータ量を小さくすることを意味します。

RTO を小さくすることは、失われるビジネス上の機会を小さくすることを意味します。

ゼロに近ければ近いほど良いのは言うまでもありませんが、その分コストが必要になります。

以下、RPO・RTO をパラメータとした構成の例です。

RPO/RTO ともに最小

完全二重化システム

RPO 大、RTO 小

スタンバイシステム（データ同期は１回/日など）

RPO 小、RTO 大

業務データのみ同期、システムはバックアップから再構築

完全二重化システムは RPO と RTO の最小化としては最も一般的で、ストレージ製品にも２台の同じ製品を仮想的に１台として使うことができる製品があります。

一方でこの場合ハードウェアは少なくとも２セット必要になりますし、そのための関連ソフトウェア、ライセンス費用や管理コスト、設置面積や消費電力、空調費用の増加も考慮しなくてはなりません。

7.2 要件 2 想定事象の検討

リカバリーシナリオを決定したら、次に対処する事象のレベル（範囲）を決定します。対処するレベルが明確になれば、おのずと必要な復旧方法と必要なインフラが明確になります。

以下、想定事象と復旧方法の例です。

表 7.1: 想定事象と復旧方法

想定レベル (範囲)	業務復旧方法（例）
装置 (部品故障)	不要（部品交換のみ）
ラック (装置故障)	復旧後に再開、またはバックアップシステムで継続
フロア/ビル (停電・火災)	復旧後に再開、または他フロアのシステムで継続
地域 (地震・津波)	復旧後に再開、または他拠点で継続

例えばフロア全体がダウンするという事象に対処する場合、少なくとも他のフロアにバックアップデータを保管する必要があるということになります。そのうえで規定した RPO/RTO を満たすための必要なインフラがある

かどうかの確認が必要になります。

また上記要件 1、2 をシステムとしてではなく、稼働する業務単位で定義することも重要です。業務の重要度や既存のインフラを踏まえ、各々のサービスレベルに見合う最適な要件を選択しましょう。

7.3 要件 3 ベストプラクティス：321 ルールとエアギャップ

要件 1、2 に加えて、バックアップの強度を高めるためのベストプラクティスについても検討しましょう。

「321 ルール」とは、「**データを 3 つ**（オリジナル 1 つ、複製 2 つ）持ち、**2 種類の異なるメディア**を採用し、**1 つはオフサイト**に保管」というルールです。

これは米国のサイバーセキュリティ・社会基盤安全保障庁（CISA、当時は同庁が運営する US-CERT）が 2012 年に提示したものですが、2024 年現在は独立行政法人 情報処理推進機構も「日常における情報セキュリティ対策」として推奨しており、一般に広く知られるようになりました。

321 ルールをストレージに当てはめると、バックアップ先となるストレージを 2 種類もち、うち 1 つは業務システムと切り離された別の環境 (例えば、外部のクラウド) に保管するということになります。

もう 1 つのベストプラクティスは「エアギャップ」という考え方です。これは業務システムやインターネットから完全に隔離された場所を指し、例えばテープストレージなどにバックアップしたうえで倉庫に保管、隔離・孤立したネットワークのシステムで保管、あるいはバックアップデータの保管を専業とするクラウドサービスに委託といったことが考えられます。

このエアギャップという考え方は特にサイバー攻撃やマルウェアなどの侵

入型攻撃への対策として有効と言われています。321 ルールのオフサイト保管と併せて取り入れると、バックアップはより強固になります。ぜひ検討しましょう。

7.4 バックアップ方式の検討

要件をふまえて必要なバックアップ方式を決定します。一般的には以下のような方式が提供されています。

- 同一ストレージ装置内

  クローニング

  複製（バックアップ）。複製のタイミングは任意。

  ミラーリング

  データ更新と同時に複製先も更新（常に同期を維持）
- ストレージ装置間

  遠隔バックアップ

  他のストレージ装置へのバックアップ。タイミングは任意。

  同期レプリケーション

  データの更新と同時に複製先も更新（常に同期を維持）

  非同期レプリケーション

  任意の周期で複製先を更新（データ更新とは非同期）

同期レプリケーションでは、サーバの書き込み指示は遠隔サイトへの書き込みが確認されたところで完了となります。このためストレージ装置間のデータ同期は常に保証されますが（常に RPO=0）、一方でネットワークの往復時間がそのままサーバ・ストレージ間の転送レートに加算されるため、日常の業務性能を維持するにはサイト間に十分なネットワーク帯域が必要になります。

非同期レプリケーションでは、データは常に一定の周期で同期されます。データの同期処理が業務処理と切り離されるため性能影響を抑え、かつ同

期用のネットワーク帯域も節約できるため性能・コスト面では有利ですが、RPO は同期の周期に依存します。

7.4.1　その他 1：カスケードレプリケーション

遠隔レプリケーションにおける性能影響についてもう 1 つ考慮すべき点は、複製元ボリュームの負荷です。特に複製元と複製先が 1 ：多の関係となる場合、複製元ボリュームには読み出しアクセスが多重でかかることになります。

このため、遠隔レプリケーションの構成ではいったんストレージ装置内でミラーリングを行い、ミラー先のボリュームを複製元としてレプリケーションする構成も見られます。

こういった要件を満たすため、複製元から中継地点を通して遠隔レプリケーション先まで一貫した同期を保証する「カスケード」機能がサポートされている製品もあります。

7.4.2　その他 2：スナップショット

ある時点のデータを保存することのできるスナップショットは、主に操作ミスの救済やデータベースである時点のデータを保証するためなどで利用される機能です。

バックアップ機能の 1 つとして扱われることも多い機能ですが、一般的なアーキテクチャは「最新版のファイル」と「更新差分（旧ブロック）」の組み合わせで実現されることが多く、その場合ファイル本体が失われると復旧できません。そのためスナップショットが有効になっているボリュームについては、事象ごとにどのように復旧できるか確認しておく必要があります。

なお、スナップショットは更新アクセスの際に差分（旧ブロック）のバックアップコピーを取るという仕組みが広く採用されており、そのため性能影響が問題視されることがあります。また安全のため多世代にわたって更新差

分を確保することは結果として容量の消費につながるため注意が必要です。

第 8 章

データ量削減

近年のデジタルデータ量の増加には目を見張るものがあります。21 世紀になりビッグデータという言葉が出てきてからもその勢いは止まることがなく、SNS 利用者の増加とその多様化や、IoT の発展などにより今後もその流れは止まる気配がありません。

ストレージはこれら大量のデータを保存するために利用されますが、このように増大化するデータを可能であれば出来るだけ少ない物理デバイス/容量に効率的に保存をしたい、更にバックアップも取得したいがその際にはパフォーマンスを少しくらい犠牲にしてでも更に小さな物理デバイス/容量に保存をしたいというモチベーションが出てきます。

その目的のために利用されるテクノロジーについて本章では説明をしていきます。

8.1 シンプロビジョニング（Thin Provisioning）

シンプロビジョニングは仮想プロビジョニングと呼ばれることもあります。このように呼ばれることからある程度想像が出来るかもしれませんが、これは仮想的にボリュームを作成する手法です。ここでいう仮想的なボリュームとは何かというと、本当は設定されたほどの容量を持っていないボ

リュームのことを意味します。

つまり、例えば本当は物理的に 10TB の容量を持っていないにも関わらず、そのボリュームを利用しているホストなどには 10TB の容量があるかのように振る舞います。

このような振る舞いをすることにより、分割損を避けながらフレキシブルな割り当てを実現し、また実際の運用状況を見てドライブ容量のコントロールを行うことによるコストメリットを享受することが可能となります。

このような例を考えてみてください。部門 A、B、C という 3 部門が共通のストレージシステムを利用する環境であるとします。それぞれの部門ではこれからの 3 年間で必要となるであろうドライブ容量をそれぞれ 100TB、200TB、300TB と予測しました。

その予想から 1 年が経ち、実際にどれくらいのデータ量が増えたのかを確認してみたところ、部門 A は 80TB も増えているのに対し、部門 B と C はそれぞれ 50TB しか増加していませんでした。つまり 1 年間で実際に必要であったディスク容量は合計 180TB でした。

シンプロビジョニングを利用しない場合、向こう 3 年間の予想を立てた時点で合計 600TB のドライブを購入してそれぞれ 100TB、200TB、300TB のボリュームを作成していた可能性が高そうです。

それに対してシンプロビジョニングを利用する場合には、それぞれ 100TB、200TB、300TB のドライブ容量があるように部門 A、B、C のホストには見せておきながら、実際には様子見として合計 200TB 分のドライブを購入するということが可能となります。

これにより 3 部門それぞれにおける未使用ドライブ領域という無駄な領域を減らすこと（分割損を減らすこと）が出来ています。また、1 年前に 200TB 分のドライブを購入しているのですが、今年も追加で 200TB を購入しようとした場合、その容量単価は 1 年前よりも低下していることが期待できるために、1 年前にまとめて 600TB 購入するよりもトータルの容量単価を抑えることができます。

このようなメリットがシンプロビジョニングには存在しますが、この機能

を実現して作成されたボリュームは、シンプロビジョニングを利用せずに作成されたボリュームに比べてオーバーヘッド処理が多く存在するために、パフォーマンス面で劣位性があります。

しかしながら、近年のストレージの処理高速化に伴い、その劣位性は人間が感じられるようなものではなくなってきているというのが現実であり、パフォーマンスに関して本当にシビアな環境でなければシンプロビジョニングの利用に関する問題はないと考えて良いでしょう。

8.2 圧縮

圧縮とは情報を符号化することにより、ビット数を減らすことを意味します。この圧縮という処理を行うことにより、データをストレージに保存する際にそのデータ容量を小さくすることが可能です。

とはいえ具体的に符号化とは何を意味していて、何故符号化によりビット数が減りデータ容量が小さくなるのでしょうか。ここでは簡単な例を使って説明します。

今「11111111112222222223333333333」という 1 が 10 個、2 が 10 個、3 が 10 個並ぶ情報があるとします。ここでは簡単のために一つの数字や記号を表現するために 1 バイト必要であったとします。そうするとこの情報をそのまま保存すると 30 バイト分の容量が必要です。しかしながら、このデータは同じ数字が 10 個並んでいるものが 3 つ存在しているので、「10*1+10*2+10*3」で表現する（符号化した）ことにします。

そうすると、これは 14 バイト分の容量で表現ができています。この状態でデータを保存することにより、元のデータの半分以下の容量で保存することができることになるために、データを圧縮したことになります。

実際にはどのような情報でも表現できるようなもっと複雑な符号化手法が取られているのですが、このように符号化を行うことによりデータ量を減らしながらも、元のデータを表現することが出来ることがイメージできると、圧縮という技術を理解しやすくなります。

8.3 重複排除

言葉の通り、重複したデータを排除することにより保存するためのデータ量を減らそうという試みを重複排除と呼びます。

例えば NAS として利用している共有ファイルサーバーがあり、そこに A さんと B さんと C さんが同じ「就業規則マニュアル.docx」という 3M バイトのファイルを保存していたとします。重複排除を行わないとこれらのファイルの保存には 9M バイト（3M バイト× 3）の容量が必要となります。

しかしながら、3 名が保存している「就業規則マニュアル.docx」というファイルの内容は全く同じなので、重複を排除して一つだけファイルを保存しておけばよいという考え方ができ、それを実現するのがファイルレベルの重複排除です。

また、上記の例に加えて「情報セキュリティポリシー.docx」と「プロジェクト A.docx」というファイルも同じ NAS 内に保存されていたとします。これら 3 つのファイルは別の物なのでファイルデータとしての重複排除は出来ませんが、3 つとも同じフォーマット、例えばマイクロソフトの Word フォーマットで記載されているドキュメントであるとします。

すると、これら 3 つのファイルの中に同じデータブロックが存在していることが期待できます。例えばドキュメントのメニューを表示するためのデータブロックは「001001110110111」という共通のデータブロックを持っているかもしれませんし、ドキュメントの空白部分は「1101010110011」というデータブロックで表現されているかもしれません。

これら共通するデータブロックを見つけて、同じデータブロックに対しては重複を排除して、ストレージの中でそのデータブロックを一つだけ保存しておけばよいという考え方ができ、それを実現するのがブロックレベルの重複排除です。

一般的にファイルレベルの重複排除よりもブロックレベルの重複排除の方が高い効果（重複率）を得ることが可能です。

なお、重複排除により複数あったデータが 1 つになったとしても、各ユーザからは重複排除が行われる前と同じく独立にそれぞれのファイルを保持しているように見えます。

第 9 章

拡張性

多くのストレージでは容量や性能が不足してきた際に拡張することができます。拡張のやり方として、大きくスケールアップとスケールアウトの 2 パターンあります。それぞれについて解説します。なお、各社のストレージによって、スケールアップとスケールアウトのどちらで拡張するかは異なります。

9.1　スケールアップ

スケールアップの概念図を図 9.1 に示します。

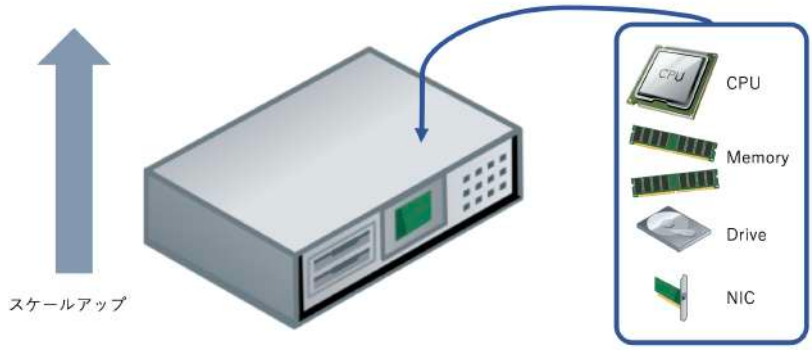


図 9.1: スケールアップ

スケールアップは、ストレージのコントローラなどの筐体に CPU/Memory/Drive/NIC などのコンポーネントを追加し拡張する方法です。このスケールアップでは、筐体の電源をオフにする必要があることが多くストレージのサービスを停止する必要が出てくることがありますので注意が必要です。また、ストレージによって各筐体にどのくらいの CPU/Memory/Drive/NIC などのコンポーネントが搭載できるかは異なります。

なお、スケールアップとは逆に縮小させることをスケールダウンと呼びます。

9.2 スケールアウト

スケールアウトの概念図を図 9.2 に示します。

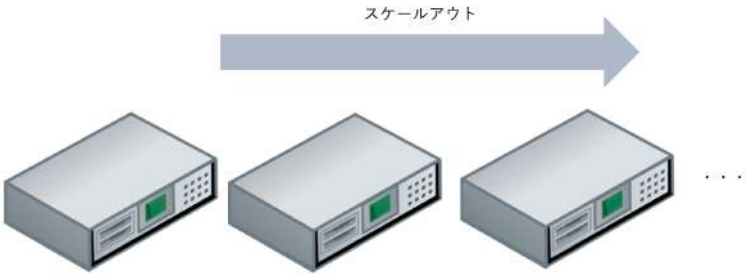


図 9.2: スケールアウト

スケールアウトは、ストレージの筐体単位で追加し拡張する方法です。このスケールアウトを採用しているストレージの多くは、筐体の電源をオフにせずとも追加できます。また、スケールアウトできる台数については、各種ストレージで異なりますが、その多くはストレージの内部処理の並列度の限界性能や筐体間のネットワーク性能により決まります。なお、特に Software Defined Storage (SDS) ではこのスケールアウトによる拡張を採用し、ストレージの筐体相当の機器を汎用サーバで構築し、この汎用サーバ単位で増設するものが多いです。

なお、スケールアウトとは逆に縮小させることをスケールインと呼びます。

第 10 章

ストレージセキュリティ

ストレージは企業の生命線である業務データを格納・保管する用途から、さまざまなセキュリティ要件が求められます。以下、ストレージの世界で一般的なセキュリティの仕組み、機能について紹介します。

なお最も基本的かつ重要なセキュリティ対策は日常の運用における「適切なアクセス制御・権限設定」であることはいうまでもありません。

- 暗号化
- 物理保護
- サイバーセキュリティ保護
- 消去

10.1　暗号化

「セキュリティ」というキーワードで、ストレージで最も一般的な機能はデータの暗号化です。データの盗難はさまざまな箇所・経路で行われる可能性があることから、暗号化もさまざまな箇所で行われています。

10.1.1 管理ネットワーク (コントロールパス)

ストレージ製品には一般的に、製品の設定や管理を行うためのインタフェースが搭載されています。アクセス許可やバックアップコピーの設定が不正に行われることを防ぐ目的から、暗号化による保護対象となっています。この暗号化は SSH や HTTPS など一般的なものが使われることが多いです。

暗号化のほかに、MFA（複要素認証）による認証の強化もいくつかの製品で採用されています。

10.1.2 アクセスネットワーク（データパス）

業務データが行き交うアクセスネットワークはもちろん、盗み見やデータの窃取などのリスクに対応する必要から暗号化が提供されています。暗号化方式は、ファイルストレージではアクセスプロトコルとして Windows では SMB や Linux では NFS に実装されている暗号化を利用するケースが一般的です。

オブジェクトストレージでは暗号化されているアクセスプロトコルとして HTTPS 通信を利用します。

また、ブロックストレージで多く利用されている iSCSI などのアクセスプロトコルでは暗号化を持っていないため、IPSec など異なる通信レイヤーでの暗号化と組み合わせて利用します。

その他、ストレージとサーバの間に挿入して暗号化を行う専用のハードウェア製品なども存在します。

10.1.3 ドライブ

データが最終的に格納されるドライブも暗号化による保護の対象となります。この場合の暗号化の目的は、設置された又は廃棄された製品からドライ

ブが盗難された際に格納されたデータを保護する目的になります。

データの暗号化処理は、大きくはストレージが製品機能として行う場合と、ドライブそのものが暗号化機能を持っている場合の２種類があります。

前者の場合はドライブへの書き込み時に暗号化を行う「インライン」タイプと、ドライブへの書き込み後にバックグラウンド処理として行う「ポストプロセス」タイプがあります。前者は確実に暗号化が行われる点で安心ですが、一方で性能影響が出る場合があります。その観点でどちらの処理方式を選択するか考慮しましょう。

後者の場合は、ドライブそのものが暗号化機能を持っている製品として一般に「自己暗号化ドライブ（Self-Encrypting Drive）」などという名前で販売されています。性能影響がほぼないとされていますが、提供されているドライブタイプ（容量や種別）は限定されています。

10.1.4　その他：暗号鍵管理について

暗号化は規定のアルゴリズムに基づいた算術処理によって行いますが、この処理には元データともう１つ、暗号化鍵と呼ばれるパラメータが使われます。この暗号化鍵が盗まれると論理的には復号が可能となるため、定期的に更新するなどの管理が必要です。

暗号化機能を持つ製品であればその製品自身が自動的に管理しているケースが多いですが、暗号化鍵を管理する専用製品も用意されています。この暗号鍵管理製品との暗号化鍵のやりとりでは標準化されたプロトコル（KMIP: Key Management Interoperability Protocol）が一般には使われています。

10.2　物理保護

ストレージによるセキュリティのもう１つの例が物理保護です。以下、いくつかの例を記載します。

10.2.1　構成保護

物理構成の不正な変更を検出する機能です。これは具体的には不正なバックドアの設置などを目的としたドライブの一時的な盗難（抜いて、データを仕込んで、戻す）を想定したものです。一般的には、抜き取られて一定時間が経過したドライブは初期化しない限りシステムに組み込まれないような仕組みによって実現されています。

10.2.2　起動保護

製品そのものの盗難・データ窃取からの保護を目的に、特定の外部デバイスとの接続を条件として起動が抑止されるというものです。

自動車と鍵の関係のように、装置の USB ポートに専用のキーが刺さっていない状態では起動が抑止されるものや、前述の暗号鍵管理サーバとの接続がされていない状態での起動を抑止するものなどがあります。

10.3　サイバーセキュリティ保護

特に不特定多数クライアントからのアクセスが想定されるストレージはサイバーセキュリティ観点の保護機能も多く用意されています。以下、代表的なものを紹介します。

10.3.1　アンチウイルス・ソフトウェア連携

ストレージ自身や格納されているデータがウイルスやマルウェアの脅威に脅かされることもあります。もちろんアクセス元クライアントでの防御が一般的ですが、クライアントのリスク低減を目的に、ストレージが外部の専用アンチウイルスサーバが連携してファイルシステムのスキャンを行うという機能も一部製品では提供されています。

10.3.2 スナップショット/バックアップ +WORM(Write Once Read Many)

主としてマルウェア対策を目的に、製品内で取得したバックアップを外部から一切アクセスができない特殊な保護領域に保管する機能が一般的になりつつあります。

この保護領域は製品の特権ユーザさえもアクセスできないようにプロテクトされており、マルウェア侵害からの確実な復旧を実現します。

10.4 消去

少し違う角度の保護機能としてドライブのデータ消去機能が挙げられます。故障や保守切れに伴い撤去される製品からのデータ窃取を防止することが目的です。

SSD では標準的にマッピングテーブルやメモリセルと言われる構成情報を消去することで論理的にデータの復元を不可能にする機能が用意されており、ストレージで提供されている機能は概ねこの仕組みを利用したものです。

HDD などの磁気ドライブでは、単にデータを削除しただけでは OS などからデータが見えなくなったとしても、確実に消去されず残留磁気によりデータが復元されてしまう危険性があります。

そのため、NIST 800-88 Advanced 方式や NSA 方式などランダムデータとゼロデータを複数回書き込むことにより、残留磁気を残さないようにし確実なデータ消去を行います。

このようなデータ消去機能は一般にはデータシュレッディングなどと呼ばれています。

なお、より一般には製品ベンダが保守サービスのオプションとして、故障

し交換したドライブを（ベンダが回収せず）ユーザに引き渡すサービスが提供されており、ユーザが自身で処分したり、別途物理破壊サービスに委託するなどしたりすることができるようになっています。

参考文献

[1] 松岡正剛, 増補 情報の歴史, NTT 出版, 1996.

[2] K. Goda and M. Kitsuregawa, "The History of Storage Systems," in Proceedings of the IEEE, vol. 100, no. Special Centennial Issue, pp. 1433-1440, 13 May 2012, doi: 10.1109/JPROC.2012.2189787.

[3] Timeline of Computer History [Online]. Available: https://www.computerhistory.org/timeline/memory-storage/

コントリビューター

本ホワイトペーパーに貢献した次の方々に感謝の意を表します。
- 横井 伸浩 (SNIA 日本支部技術委員会)
- 大内 敦夫 (SNIA 日本支部技術委員会)

ストレージ初学者のためのホワイトペーパー

2024 年 6 月 1 日　初版第 1 刷　発行

著　者　伊藤佳治、上原勇太郎、菊地孝浩、坂下幸徳、中村隆喜
監　修　SNIA 日本支部技術委員会