Compartmentalized Cybersecurity & Incident Recovery

October 2021

Fred Bonner
EchoLeaf Systems

Peter Guglielmino
IBM Corporation

Dr. John Hoehn
Life After Television

Table of Contents

Foreword

Today ransomware and other cyber attacks have increased in number and severity, posing a serious threat to business, national security, and indeed to our way of life. Due to this new threat, organizations of all sizes most often lack a coherent response plan for cyber attack or breach. Conventional disaster recovery (DR) plans are often not suitable for recovery from cyber breach, since they focus on outages and restoration rather than the clean-up of malicious destruction.

While an enormous amount of attention and resources have been spent on prevention and detection of cyber threats, cyber security professionals agree that no level of prevention can guarantee 100% success. Winning the cyber battle against criminal hackers 99 out of 100 times might seem like a good winning percentage. But what happens on day 100? That's when recovery is needed.

This paper responds to the urgent need to develop a framework and working architecture for data protection focused on rapid recovery from cyber incidents, which can now be done affordably, reliably, and with relative ease using existing technology.

In response to this threat, in May 2021, President Joseph Biden authored an Executive Order tasking government agencies with developing a cyber attack response protocol. Private and public companies worldwide are creating similar plans.

Additionally the government, through the National Institute of Standards and Technology (NIST), has offered numerous sets of guidelines for protecting data, including NIST SP 800-184 *Guide For Cybersecurity Event Recovery* (2016) and NIST SP 800-209 *Security Guidelines for Storage Infrastructure* (2020).

Though the NIST documents provide a set of recommendations, it is the responsibility of experts in the IT community to translate the recommendations into practical solutions for the real world.

The urgency of the moment is new. The destructive capabilities of cyber attacks are now more pronounced, visible, and menacing. Mature and robust technologies exist today to ensure, in almost all cases, a rapid and secure recovery from a cyber breach. This document outlines a proposed playbook of requirements, explains the technical rationale for each citing NIST recommendations and ISO standards, while providing concrete examples of how to implement the solution.

With this compartmentalized architecture, organizations of all sizes will be far better prepared and far more resilient to meet these threats.

Executive Summary

While most efforts to counter ransomware and other malware are focused on prevention, it is imperative that every organization be prepared to recover key data and systems after a destructive or system-compromising cyber attack.

Different organizations will have different requirements for system restoration, but at a minimum, architectural priorities require that the proposed blueprint can counter the major threats and attack vectors of today's malware: i) Malicious encryption ii) Theft of data for extortion or espionage iii) Malicious technical disruption.

The proposed architectural requirements are not aspirational, but rather can be deployed with off-the-shelf, standard technologies made by major manufacturers. Technological advances now allow a compact and effective architecture for pod-level, compartmentalized recovery that will meet the urgent threats posed by ransomware and other malicious exploits. The design leverages enterprise grade technologies in a hybrid environment.

Applying the principles of Zero Trust Architecture (ZTA), the goal of rapid recovery dictates that compartments of data activity should be isolated much like the compartments on a ship, where a data or systems breach in any one compartment will not affect other systems. The ZTA principle of assuming a breach means other parts of the network may also be compromised, which can include central recovery resources. Recovery pods should be created such that, if a failure occurs at central systems, individual compartments or pods can be quickly restored with all assets needed for pod recovery contained in the pod.

NIST recommendations can be divided between functional and non-functional requirements. A non-functional requirement would be something like "document all procedures." A functional requirement is more prescriptive, for example, the solution must have the ability to create multiple copies of each data set.

This architectural design supports the relevant NIST functional recommendations plus other tested security principals. In all cases, there are many degrees of freedom to adapt the architectural requirements to real-world needs. A school district, for example, will have different requirements than a naval base – yet many of the same principles will apply. In all cases, however, meta-requirements determine that the proposed architecture be flexible, low cost, easy to deploy, durable, and highly reliable.

One size will not fit all. The proposed architectural blueprint is not suitable for high-transaction systems or time-sensitive mission-critical real-time systems. However, a vast number of systems and subsystems in most organizations will find the architectural recommendations and requirements useful and appropriate to their mission.

The architecture is non-denominational—no specific products, brands, or companies are mentioned. However there is a bias towards reliable, proven technology that, while not new in any conventional sense, has advanced greatly in sophistication, technical capability, with affordable costs.

While data storage is an essential part of our cyber infrastructure, like all advancements, it is incumbent on us not only to deliver high level functioning, but also to protect against things that can go wrong. Cars get airbags and seatbelts. Office buildings get sprinklers. Kitchens get fire-extinguishers. Essential data can be lost through simple mistakes, but now also through malicious intent. Basic backups and disaster recovery plans were good enough when the only problem was either mistakes or natural disasters. The malevolent nature of cyber attacks in the present environment requires a renewed architecture to meet the unique requirements of after-breach recovery.

Succeeding in this critical design effort is essential. The costs for failures? Business disruption, loss of revenue, infrastructure and supply chain breakdown, remediation costs, regulatory breach, loss of institutional and organizational history, reputational loss, and jeopardizing national and international security.

Our experts are telling us cyber breaches are inevitable. Knowing how to recover, quickly and elegantly, is imperative.

PART 1

Introduction: Architectural Crossroads for Data Preservation

Breaches are Inevitable

Cyber war is here, though not publicly declared. On the one hand, technology is proliferating at a dizzying pace. On the other hand, criminals, some supported by well-funded nation-state actors, wish to destabilize Western society through the destruction and perversion of that technology through ransomware, data theft, espionage, and outright system destruction.

Because attacks come relentlessly every day, political leaders are demanding new technical architectures that will provide both defense against malefactors and resilience and recovery in the face of attack. In fact, NIST SP 800-209 (October 2020) calls out the distinction between recovery from non-malicious or routine needs, and recovery from cyber attack.

On the defense side, Zero Trust Architecture or ZTA now provides a roadmap for building protected systems both in local environments and in the cloud. ZTA states that no perimeter protection is to be trusted, breach is always assumed, and systems must be compartmentalized such that a breach in one area does not create a breach in others.

Even with these principles intact, the high-value nature of data means computer systems of all types are still the targets of feverish hacking by international groups of criminals. Experts around the world agree that destructive breaches are inevitable. There is no equivocation; breaches will be a fact of life. The phrase "not if, but when" is literally axiomatic among technical professionals when referring to large scale, destructive cyber attacks.

If destructive security-compromising breaches are inevitable – successful recovery is essential.

What Are the Threats?

Systems must be built that are resistant to cyber attacks, and preparations must be made for speedy recovery. Things like seat belts, bomb shelters, and safe rooms should be rarely used, but most are deployed in preparation for trouble. Designing for cyber recovery is no different.

The primary threat vectors for adversaries today include: i) Malicious encryption – the primary tool of ransomware; ii) Theft of data for extortion or espionage. This includes planting exploits for long-term snooping; and iii) Malicious destruction – exploits that can damage equipment or infrastructure.

Defenses include, but are not limited to, enhanced password protection, two-factor authentication, network sniffers, various network measurement and intrusion detection systems, user education, and with ZTA guidelines, constant monitoring of who is in the system, and what they are doing. These requirements and solutions exist for cloud, on-premise, and everything in between including smart phones, laptops, and countless other devices. While many of these defensive systems are automated, they are still reliant on configurations, settings, and sometimes monitoring by humans, and are therefore vulnerable to human error.

The questions of how to respond to a breach, how to recover, how to become resilient in the face of the most damaging attack, should be at the center of system design. However, with the well-funded sophistication of attackers and ever-increasing availability of tools with malicious intent, the fight takes on the character of a wrestling match. There will be take-downs on both sides.

Architectural Choices for Recovery

The challenge is to acknowledge the deficiencies in the present architectural designs and to create systems that will respond to new cyber threats by providing resilience after an attack.

There is a saying among technical architects – *most design mistakes happen on the first day.* As we reimagine our recovery posture, this is day one. What are the design factors that can guarantee a comprehensive, speedy, reliable recovery after a cyber attack? It is a question not just for every government, and every public and private entity charged with protecting its data, but also for each individual person.

The problem is that the prevalent data technology build patterns are centralized and built for performance, control, and efficiency, not intended to withstand after-incident cyber recovery. The cloud is built for standardization and rapid scaling, but it is reliant on the internet. Primary systems today have been built for perimeter defense and trust relationships – not to defend against a daily onslaught of increasingly sophisticated cyber threats and not to facilitate rapid rebuild during after-attack recovery.

Cloud-based architectures have been successful. But it is important to note that many, if not most, systems today work with both cloud and on-premise architectures – "hybrid" is the term of art.

Why would we therefore try to deploy exclusively cloud-based systems to construct our methods of recovery and resilience? Or will we, in addition, also consider hybrid designs, providing the best technology we can for bunkered, compartmentalized, on-premise, recovery?

Including hybrid design is important for an essential reason. Hybrid architectures that include on-premise recovery nodes, in many use cases, can be made vastly superior to cloud-only solutions.

Hybrid Architectures

On-premise recovery nodes will have several clear advantages over only remote or cloud-based options.

First, local recovery assets will have no dependency on the internet and can be easily and rapidly accessed. If recovery assets are large and remote, beyond the 100s of GB range (let alone many TBs), the time to download and make assets available to a local system over the internet will be considerable. *In the most extreme circumstances, there may not even be an accessible and trusted internet to transport recovery resources.* Certainly, any location or region that is internet constrained will have problems.

The second advantage is that, by using enterprise level tape archiving for local assets, recovery assets can be made impervious to any attempts at malicious encryption or corruption. Tape is an append only medium, and therefore files cannot be overwritten (if properly configured.[1]) In addition, even within a tape library, tape is air gapped and files on tape cannot be altered or corrupted. Alternatively, volatile connected systems can be destroyed in a flash of malicious encryption commands.

Cloud vendors will claim that data there can be fully encrypted, duplicated to tertiary sites in the event of catastrophic loss, compartmentalized, made immutable, and versioned so that in the event of a breach you can find files from previous days, weeks, or months.

Today tape, out of the box, can do those very same things in a standards-based environment, with no enhancements and no extra cost. In addition, tapes are also portable, durable, and highly resistant to EMP (electro-magnetic pulse.) Tapes with data intact survived the Columbia Space Shuttle explosion, where other data retention technologies failed[2]. And with the density of tape today, a small assembly of 5 rack-units will hold over 750 TB. It would take 52 days at a fully saturated 133 Mbits/sec to transport that data over the internet.

A final point in favor of tape surviving disaster – architectures can be designed where, in the face of total loss of all connected data and systems within a pod or compartment, non-volatile tape can hold *all* the digital information needed to reconstruct the pod, including needed encryption keys.

[1] Tape Configuration Options: There will be numerous options in terms of how enterprise digital tape is deployed. In the past, as a block-based storage mechanism, tape storage was dependent on an external file index with knowledge of any particular tape's data block structures. Using ISO standard LTFS formatting for tape, all tape deletions are logical rather than physical, and the block index is, within that standard, a physical part of the tape. Therefore all deletions can be rolled back to a previous index. At that point, the only danger to data on the tape would be either physical destruction, or forced formatting, a function which can be either encrypted or removed as a system function.

[2] Weiss,Todd,
Computerworld, 3 April 2003, "Old technology offers new answers in Columbia crash probe",
https://www.computerworld.com/article/2581141/old-technology-offers-new-answers-in-columbia-crash-probe.html

It is easy to invent scenarios where either cloud or on-premises architectures will provide adequate recovery. It is also easy to invent scenarios where either one may fail. Tape is an important part of our technological defense system. Tape technology has progressed to the point where it can be successful where cloud-based architectures will fail or be inadequate. For systems large and small, hybrid systems including both cloud and on-premise tape may be best to meet the urgency of the moment and beyond.

There are some obvious architectural examples for creating compartmented, redundant areas of recovery. For example, standby power for data centers: Relying on public utilities for 100% up-time is not best practice. Localized standby generation is required to ensure uninterruptible operations. Substitute Public Cloud for Public Utility power and you are compelled to consider localized recovery capabilities to address system and data outages.

The foundation of a compartmentalization pattern implies acceptance of a hybrid or an on-premise operations model. The core sources of recovery *must* be local and under complete local control. If an enterprise is 100% internet dependent, cloud-based, public cloud, and/or out-sourced, local on-premise recovery capabilities, like those described here, will be ineffective unless those non-local resources are backed up to an on-premise, compartmentalized system.

The architecture needs to be well planned and executed. The Texas Power Grid collapse in winter 2021 offers some important lessons to consider. The Texas Grid was isolated and compartmentalized - however poor planning and bad design of alternative energy generation resulted in complete grid failure. So along with compartmentalization there needs to be IT infrastructure capacity planning, probably workload repatriation required from cloud or other data centers, as well as periodic recovery testing scheduled to ensure recovery will be successful.

The document from the National Institute of Standards and Technology (NIST) from October 2020 entitled "Security Guidelines for Storage Infrastructure" available at https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-209.pdf provides a comprehensive set of recommendations for securing storage infrastructure. However, while guidance and general recommendations are set, the document falls short of recommending a specific architectural plan.

The following compartmentalized resilient architecture, while allowing many degrees of freedom in its implementation, contains actionable requirements to create recovery pods that will withstand new requirements created by the prospect of cyber attacks.

PART 2

Definitions, Principals, and Priorities of a Compartmentalized Recovery Architecture

Definitions

Threat Vectors

What can fail during a breach? We have learned what a full breach can entail, and our recovery mode needs to assume that any of those things can occur. The broad categories of damage from a breach are[3]:

➔ ***Malicious encryption*** of all connected devices and services including backups, virtual machines, and connected cloud services or backups.

➔ ***Theft of data***, often exfiltrated before awareness of the breach. The purpose of data theft might be extortion or espionage.

➔ ***Malicious disruption*** of equipment or control systems.

Adaptive Recovery

Flexibility based on restoration requirements is essential. Creating a system that can be restored quickly after a breach is possible. We can create very high levels of fault tolerance. However, this level of security comes with a cost. Think of a fully redundant secondary system with hot failover. Sometimes this may be necessary, but not always. The desired deployment of the restoration architecture needs to be tuned to the requirements of the pod's function and the agreed risk / service level to the organization.

Recovery Assets

Recovery assets constitute the technical artifacts (e.g., backups, system images, programs) needed to restore the pod to a full recovery, however that has been defined. What is essential to understand about this architectural approach is that, even assuming a pod-level breach, *recovery assets can be protected from malware exploits while they are being maintained in the pod.* It is no longer necessary to maintain all backup media entirely offline to keep them safe from common exploits. True offline assets *may* be necessary in the event of catastrophic loss; but those offline assets can be maintained in a way that pod-restoration – even in a new location – can be accomplished efficiently.

Key Recovery Principles from ZTA

The following points are essential to creating an architecture that can quickly and successfully be recovered. Zero Trust Architectures are the defining model today for creating secure computing

[3] NIST SP-800-209 Section 4.7 Isolation

environments. These general principles are discussed in NIST SP 800-209 Security Guidelines for Storage Infrastructure.
(NIST SP-800-209: IS-SS-R1 Separation of Storage Systems. [4])

Perimeter Security Is Not Viable

Perimeter security is no longer trusted. You cannot simply secure a technical perimeter and assume that everything inside of it is safe.
(NIST SP-800-209: IS-SS-R1 Separation of Storage Systems [5])

Breach Assumed

Always assume a breach – therefore every connection between any entity must be verified.
(NIST-SP-800-209: IS-SS-R2 – Separation of Management systems.[6])

Keep Recovery Assets Local

Understanding that an entire pod or compartment, however it is defined, can be compromised by a breach, recovery assets, those items needed to recover the pod, should remain close to the source for needed restoration. In a breach, network resources may be compromised, and internet and cloud access (private or public) may be nonexistent or compromised.

Compartmentalize Recovery Needs

Recovery Zones in a ZTA are like shipboard compartments. A breach should be contained within the compartment. And therefore, recovery assets should be available in the compartment.

Assuming a breach means compartmentalization is a necessity.

[4] *IS-SS-R1 Separation of Storage Systems:* The standard recommends "separated" storage – without definition. The important aspect of said separation is that it should be separated from production storage. Any type of storage that can be overwritten in the process of a production activity would, by definition, not be separated. Similarly, storage that is physically separated or airgapped to the point of being physically removed from any production activity is also separated. Enterprise tape, however, configured using the LTFS (Linear Tape File System) ISO standard, cannot be overwritten in any conventional sense, for two reasons. First, tape is an append-only medium, and therefore files once written to tape cannot be changed. Tape does, however, allow logical deletions and logical overwrites – meaning it can appear that a file is deleted or changed. However, *the original file still remains on the tape and cannot be changed.* Using capabilities outlined in the standard, LTFS tape can roll back its indices to any previous time, meaning any logical deletions or logical over-writes can be undone. This gives data written to the tape the character of being immutable. While it also fulfills the functional requirement of being separated from production systems in that, once files are written, they cannot be *physically* overwritten by any production process. Additionally, if there is a requirement to create replication copies, tape can accomplish this easily. And those replicated copies can, if needed, be physically placed offline.

[5] *IS-SS-R1 Separation of Storage Systems: ibid.*

[6] *IS-SS-R1 Separation of Storage Systems: ibid.* Using a dedicated Storage Subsystem Caching Server (SSCS) in the architecture, where production systems only interact with that server through SAMBA or NFS protocols, the management system for the storage is separated from the rest of the production environment.

Unfortunately, compartmentalization flies in the face of large monolithic technical infrastructures that have been constructed for centralized efficiency. Security comes by creating pod-level safe rooms, recovery nodes, or recovery pods, that can both be isolated in the case of an assumed breach and also recovered and brought up to full efficiency in any time period, however short, as determined by the pod's operational importance.

This is not to say that the recovery pod cannot participate in larger enterprise initiatives. It can, and inevitably will. However, for the sake of after-breach recovery, the pod needs to be more compact, and to hold the keys to its own recovery within its specified compartment. Large organizations can continue to operate as they always have. No operational changes are required other than to create tertiary restoration pods based on the ZTA model. Central backups can still exist (if they are considered safe) but they must be coupled with pod-level resilience.

Architectural Priorities

Based on the principles established by Zero Trust Architectures and NIST SP 800-209, the following architectural priorities emerge.

Threat Protection

Protections against ransomware and other exploits become the priority. The architecture must be able to respond to malicious system encryption, attempts at data theft and/or espionage, and exploits with the intent of malicious disruption.

Recovery Optimization

The architecture should be designed such that rapid recovery is the priority. The recovery process is to restore full function to the system as a whole, but to do so in a way that essential data or services can be restored most quickly. Specific Service Level Agreements (SLA) determine how recovery is defined for the pod.

Recovery Time Depends on Mission

In simple terms, when a system that is working becomes compromised or made non-functional after an attack, it must be restored to its previous state in as little time as possible with as little data loss as possible. RTO refers to the Recovery Time Objective which will be different for each secured entity.

If the services provided by a pod can be down for a day or a week, that determines the outer boundaries of the restoration requirement. If the service provided by a pod can only be down for a few hours, then the recovery requirement needs to match. As a rule, there will be more redundancy, cost, and overhead associated when recovery requirements are more aggressive.

Levels of Recovery

After a breach there will always be data loss. Even within a compartmentalized pod, in-flight data cannot be 100% recovered because it cannot be assumed that the data is not corrupted by malware as it is being written to a backup device during the breach. Depending on how accurately the boundary between attack and normal operations can be determined will determine the amount of data lost and how much potentially manual work must be done to replace the data lost between the time the attack started and the time when a 100% valid clean backup or transaction log files were created.

Therefore, the term "as little as possible" is aspirational. There will always be data lost after a breach that needs to be recreated. Each recovery entity needs to establish an RPO – Recovery Point Objective – to determine how much data loss is acceptable after a breach.

PART 3

Meta Requirements

When evaluating specific solutions that adhere to the technical requirements, the following meta requirements should be given weight.

a. **Affordability:** Understanding that compartmentalized ZTA architectures might require many installations, cost to purchase, install, and maintain is important.

b. **Ease of installation:** Pod level rather than centrally installed restoration systems should necessitate that installation is simplified.

c. **Ease of use:** Restoration systems should be easy to use and maintain.

d. **Flexibility:** Similar systems should be able to support a range of use cases determined by the size of the data stores to be protected, the length of data retention, the needed restoration window, and the pod location and availability of technical support.

e. **Future-proof:** Most threats today are understood. Tomorrow will bring new threats. To the greatest extent possible, the restoration system should be future-proofed. For example, new quantum computing techniques may render conventional RSA encryption unworkable. How will the architecture address this fact?

f. **Simplicity:** The system should be as simple as possible while still meeting requirements. Performance and technology need to match the mission of each restoration pod and provide flexibility for expansion as security thresholds increase over time. As a general principal, high complexity creates both cost and unknown vulnerabilities.

g. **Single-device access to restoration assets:** ZTA requires that trust relationships are continually tested. Therefore, restoration assets should be accessible through a minimum number of connections, one being the best number.

h. **Standards based (NIST / ISO):** The Executive Order requires compliance to technical standards. Technologies that adhere to NIST and ISO standards should be preferred.

i. **Technical maturity:** As a companion to low technical complexity, technology that is technically mature will probably have fewer problems.

j. **ZTA compliant:** While ZTA is a set of principles, solutions should be evaluated against that ideal.

PART 4

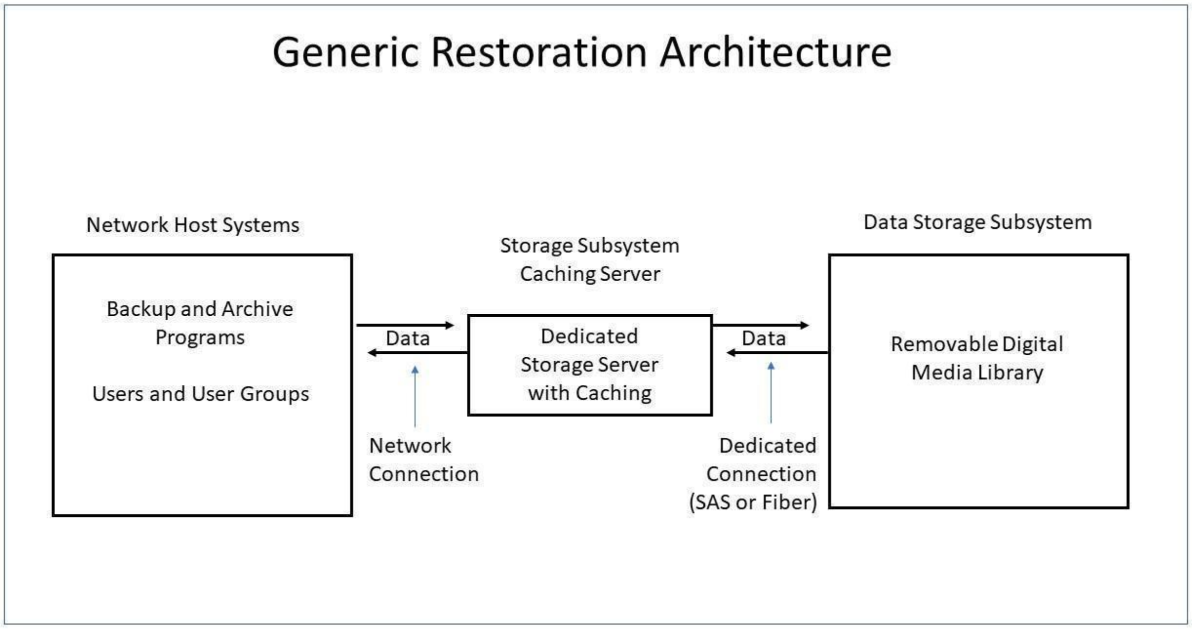
Generic Architecture, Assumptions, and Functional Requirements

Given all the above considerations and meta requirements, the following generic architecture is being proposed as a workable blueprint. For each functional requirement, available technology is matched accordingly.

This architecture is not aspirational. Rather it has been designed with essential components and functions that have been fielded with standard technology made by major US manufacturers.

Generic Restoration Architecture

The left side of the diagram, "Network Host Systems" represents the collection of systems that do work within the resilience pod. This would include workstations, remote devices, databases, and software that collects and manages backups and restoration.


Generic Restoration Architecture
Network Host Systems
Backup and Archive Programs
Users and User Groups
Data
Network Connection
Storage Subsystem Caching Server
Dedicated Storage Server with Caching
Data
Dedicated Connection (SAS or Fiber)
Data Storage Subsystem
Removable Digital Media Library

The server in the center of the diagram, the "Storage Subsystem Caching Server," provides the single point of contact for the network through standard network connections. Ideally only backup and archiving applications will have access to this server, but in a practical sense there may be many users and processes with access to this server. The general operation of this architecture allows files from the Network Hosts to be written to the Storage Subsystem Caching Server at network speeds. However, once on the Caching Server, files are expeditiously moved to the Data Storage Subsystem – and then stubbed and deleted on the Caching Server. Transfers updates, and

authorized deletions to the Caching Server can occur with any frequency required by the backup and restore programs in use.
(*DP-SS-R10, DP-SS-R11*[7])

The files on tape will be encrypted, and then stored on non-volatile media. (Details on the process below.)

The right side of the diagram, the Data Storage Subsystem works in conjunction with Storage Subsystem Caching Server, but in *ZTA they can be considered a single device*. The Storage Subsystem Caching Server and the Data Storage Subsystem should be connected through dedicated communications – either SAS or a dedicated fiber switch. These connections will be configured without network or external access.

Many organizations and departments will fit into this generic architecture. The Network Host Systems box can have infinite variety. The constant in the architecture is that the server and media library, the center and right in the depiction, provide *all the needed recovery assets in the event of a breach*. The caveat of course is that there may inevitably be some data lost between the start-time of the attack and the last valid backup. How the architecture operates is described below, with references provided against NIST SP 800-209 recommendations. (A conformance matrix is included in Appendix 1).

Assumptions

Some architectural components and decisions are assumed in creating this model. Known technologies can answer the requirements fully.

Linux Server

Linux is ubiquitous, well supported, and statistically poses far fewer security risks than Windows servers. It is the obvious choice. The Storage Subsystem Caching Server should run Linux. In addition, standard Linux configurations will allow authentication and data access control, and audit logging.
(*AL-SS-R1:R5, IS-SS-R1, IS-SS-R2, IS-SS-R3, IS-SS-R6, IS-SS-R7, RA-SS-R9, RA-SS-R11*, many of the encryption requirements *EN-SS R1-R6*[8].).

[7] DP-SS-R10, DP-SS-R11 Point-In-Time Copies: The Storage Subsystem Caching Server (SSCS) receives any copies or snapshots as determined by the backup software. And the SSCS receives them at network speeds. However, on a schedule set by the operators, the SSCS moves copies and snapshots to the tape archive.

[8] Linux Server in Architecture: AL-SS-R1:R5, IS-SS-R1, IS-SS-R2, IS-SS-R3, IS-SS-R6, IS-SS-R7, RA-SS-R9, RA-SS-R11, and many of the encryption requirements EN-SS R1-R6.

AL-SS-R1:R5; Audit logging, time sync (Network Time Protocol), log maintenance, log level, log maintenance, and SIEM (Security Information Event Management) can all me managed via the Linux OS, where the Linux server is the single point of contact with the Storage Subsystem.

IS-SS-R1:R3; Separation of storage systems, separation of storage management system, and access restrictions can all be accomplished with this architecture using the topology of a single Linux server providing the single point of contact with the Storage Subsystem.

Removable Digital Storage Media – Digital Tape

To fulfill the rigorous requirements of being breach resistant, conventional connected storage will simply be too vulnerable. In a breach, all connected media will be suspect. A library device supporting removable digital tapes will offer the data volume and flexibility to fulfill the needed requirements. Tape standards like LTO (Linear Tape Open) and LTFS (Linear Tape File System) are ISO standards and supported by many vendors. Tape today is highly prevalent in deep storage architectures for Amazon, Microsoft, Google, and many other vendors. It is mainstream technology and accounts for at least a tenth of the global storage business. (*IR-SS-R2, IS-SS-R4, IS-SS-R8, IS-SS-R10* [9])

IS-SS-R6:R7; The single Linux server in the architecture can be configured to perform only the necessary functions required by Host system access and Storage Subsystem access. All other unnecessary services can be disabled as required. Using this topology and a mediated approach to the Storage Subsystem, hosts or applications are *never mapped directly to the Storage Subsystem*.

RA-SS-R9; The requirement to separate restoration and data and applications is accomplished using the Linux server as a mediating device between host applications, host application data, and the Storage Subsystem.

RA-SS-R11; Cyber hygiene of data copies is accomplished using the Linux server as a mediating device between host applications and host application data and the Storage Subsystem.

EN-SS-R1:R6; Using the Linux server as a mediating device between host applications and host application data and the Storage Subsystem, the Linux OS can provide multiple modes of encryption, including end-to-end encryption, depending on the pod or compartment's requirements.

[9] IS-SS-R4: Although the requirement calls for "Off-site storage" – which *can* be accomplished with tape storage – it is important to focus on the reason for off-site storage and to understand that a properly configured library of tape cartridges, on-site, with an off-site mirrored set of tapes, can accomplish the same requirements. The recommendation states in regard to off-site storage: "This ensures that if the attackers have physical access to the production site or manages to compromise the physical site, they would not be able to access or compromise the cyber attack recovery copies." There are really two concerns here. If it is thought that attackers may in fact physically attack a site, then secondary off-site copies are essential. This also affords protection against catastrophic loss (a more traditional Disaster Recovery use case.) In the second instance, as long as storage is entirely encrypted, and the encryption keys are held offsite, then even access to the physical storage would be useless to attackers. Each use case is different, and system administrators must balance the value of having "cyber attack recovery copies" close at hand, against the inconvenience of keeping ALL recovery copies off-site.

IS-SS-R8: Tape cartridges, even in a library, are inherently air-gapped since tape is an append-only medium. The ISO standard Linear Tape File System (LTFS) allows for index rollback such that even if a file is ostensibly overwritten, either maliciously or accidentally, the older copy *still exists and can be recovered*. In this way, tapes that are in their home slots in a library provide an air-gap.

IR-SS-R2, IS-SS-R10: Tape, using LTFS, is immutable storage. One could always invent scenarios where any ostensibly immutable storage can be destroyed. Taking a sledgehammer to either a tape or a hard drive can destroy data. A powerful EMP blast will scramble all data in even the largest disk array (although tape is in fact far less impervious than disk to EMP radiation.) The point is that by any reasonable definition, properly configured tape is immutable.

Recovery Node Architecture

The Recovery Node, including the server and tape subsystem, should be designed as a single ZTA device. It is essential that the tape subsystem is a single use for the pod and is directly connected to the server with no other access, except possibly for service modules with no direct contact with the discrete storage devices. (I*S-SS-R1*[10])

Functional Requirements

The list of requirements below has been created in response to cyber attack threat vectors. In the case of a breach, the recovery architecture will be immune to the cyber threats introduced in the breach, or at the very least can quickly remedy any problems the threats cause. Additionally, requirements should offer flexibility to meet various mission profiles.

a. Non-volatile Media

To act as a viable storage media in the recovery pod, tape is non-volatile. It is an append-only mechanism and once written to, cannot be overwritten if properly configured [11]. Although not strictly compliant with WORM standards (Write Once Read Many) it provides many of the same features with greater flexibility and at lower cost. Most essentially, digital tape is not overwritable, and therefore it is impervious to the first threat of ransomware – malicious encryption. Since files written to tape are append-only, *they cannot be maliciously encrypted.*
(IR-SS-R2, IS-SS-R10[12])

b. Air-gapped

Digital Tapes are inherently air-gapped, even when they reside in a tape library. The normal resting place for tapes in a library is their "homeslot" – an inert plastic box to hold the tape. When in the home slot, the tape cannot be read or written to. If an application wants to write to the tape, the tape is moved through robotics to a tape drive, where writing activity can occur. However, all files previously written to the tape are safe from being overwritten.
(*IS-SS-R8*[13])

- Tapes are stored in inert containers and not available for writing unless the system robotics move them to a tape drive. Once in the tape drive, previously written files on the tape *cannot be destroyed or encrypted.* Tape file deletions are logical and not physical. The ONLY way to remove files from a tape is to entirely reformat the tape – and this feature can be disabled on a library.

- Tapes in a library, but not in a drive, are said to be "Nearline." But they offer most of the protections afforded by physically offline, co-located tapes – with many more advantages.

[10] IS-SS-R1: The proposed architecture separates the recovery storage system from other production systems.

[11] See "Tape Configuration Options" footnote 1.

[12] IR-SS-R2,IS-SS-R10: See footnote 9.

[13] IS-SS-R8: See footnote 9.

c. Nearline Tape Storage

The tape library should be configured to allow tapes to be stored in inert homeslots, and not in a device that can read or write from the tape.
(*IS-SS-R8*[14])

d. Removable Media

The tape library should be configured to allow physical tapes to be removed from the library. If the Storage Subsystem requires files from tapes not physically in the library, the system should identify the needed tape to be reinserted. See section below on "File or Object Redundancy" for a method where nearline and offsite tapes can function in concert.
(*IS-SS-R5*[15])

e. Industry Standard Formats

The most popular digital tape today is LTO, Linear Tape Open. It has been supported by a multi-company industry coalition. Fully half of all digital tapes deployed in the world today are LTO. If the actual formatting of the LTO tape is LTFS – the Linear Tape File System – that, too, is an ISO standard.
(ISO/IEC 20919:20)

f. Rollback Capabilities

If either through mishap or malice, files on tape are ostensibly overwritten, the overwritten file is never destroyed. Tape utilities offer Rollback capabilities (built into the LTFS – Linear Tape System – ISO standard) to be able to retrieve all files on a tape, even those that are seemingly overwritten.
(IS-SS-R10[16])

g. Encrypt Digital Media

To respond to threats of data theft or data espionage, it is essential that data written to tape be encrypted. Tape encryption is an established technology and is accomplished very quickly via hardware on the tape drives. In this architecture, tapes are encrypted with symmetric encryption.
(*EN-SS-R7*[17])

h. Encrypt Keys

The encryption keys on each individual tape are encrypted with asymmetric encryption, and therefore need a key server application or authority to decrypt the key. The transfer of keys should be over encrypted channels.

[14] IS-SS-R8: See footnote 9.

[15] IS_SS-R5: Offline tape copies can be configured such that they contain an "Independent, full baseline copy." Advanced configuration can allow the entire storage pod or container (from a software standpoint) to be rebuilt from the data on the tapes only.

[16] IS-SS-R10: LTFS standard allows for rollback capabilities, such that any tape can be rolled back to a previous state.

[17] EN-SS-R7: Setting encryption for data in transit is considered a standard network function, and can be supplied through the RedHat operating system together with various switch and IP settings.

(EN-SS-R8[18])

i. File or Object Redundancy

Depending on the mission requirements and use case, redundant copies (replicas) of the encrypted files might be required – or copies might be required offline (encrypted, but offline.) For protection against catastrophic loss, the system should be able to mirror content remotely. Creation of those replicas can be made over encrypted connections. Housekeeping on replicas can be managed centrally or remotely.
(*DP-SS-R5, DP-SS-R6, DP-SS-R7, DP-SS-R9*[19])

j. Full Server Recovery from Tape Media

If the tapes in the tape library are non-rewritable and non-volatile, they will be the sole survivors of a comprehensive breach. In this case, it is essential that the tapes contain all of the system files needed to reconstruct the Storage Subsystem Caching Server. Once that server is wiped and restored by the tapes, all other systems in the resilience pod can be restored expeditiously, since the main non-volatile storage system has been brought online.
(*RA-SS-R2, RA-SS-R3* [20])

k. Storage Volume

The present data volume size, uncompressed, in the newest announced version of LTO (LTO9), is 18TB per tape. Therefore, a small library only 3 rack units tall can hold 40 tapes, for over 750 TB. LTO tapes have increased in density every three years for the last 15 years, and the present roadmap will have single tapes exceeding 60TB within the next decade. Commercial tape libraries span from 100s of TB to 100s of PB. Throughput from a single tape now exceeds 500 MB/sec (or 30 GB per minute.)[21]

[18] EN-SS-R8: The physical act of storing symmetric keys with asymmetric encryption can be a "data in transit" encryption process.

[19] Replication and Mirroring:
DP-SS-R5: Replicated and Mirroring can create exact duplicate file structures of primary tapes on tapes at tertiary sites – multiple sites if required.

DP-SS-R6: Minimal trust can be created between replica or mirrored sites.

DP-SS-R7: Data transfer between replica or mirrored sites can be encrypted with standard protocols.

DP-SS-R9: Housekeeping on replica or mirrored sites can be accomplished in tandem with the same functions on the primary set of tapes.

[20] RA-SS-R2, RA-SS-R3,: Data recovery from a local, secure tape architecture can include a) All needed date assets, b) Tiered based on restoration priority (or other criteria), c) Simplification of federated consistency across applications, d) Matching all speed requirements, since the restoration assets are local and will not require internet access or long WAN runs, that may be compromised in the case of cyber attack.

[21] https://www.lto.org/

l. Multiple Restoration Use Cases

Within this architecture there are many options to add throughput speeds, restoration speeds, and storage capacity. If the 80/20 rule applies, most resilience pods will fit comfortably within a set of moderate use cases. However, the system should expand or shrink to meet most mission profiles.

m. Forensic Analysis

Restoration of assets should be restored to and mediated by the caching server, since it is a secure location within the ZTA compartment. Recovery assets should be transferable from tape into a protected area (folder/sub-directory) on the caching server for forensic inspection and scanning before being moved into the production environment. In this way restoration assets will be analyzed for known malware within the security of the ZTA compartment.
(*IR-SS-R3, RA-SS-R11* [22])

[22] *IR-SS-R3, RA-SS-R11:* A major difference between recovery after a cyber attack, and non-malicious recovery, is that many assets may need to be inspected for malicious code before being returned into production.

PART 5

Use Cases and Threat Scenarios

Use Cases

Use cases need to be evaluated based on their mission. The best use cases for the proposed architecture will be where data tape is optimized for backup or archive within a discrete ZTA environment, storing data with low to moderate retrieval requirements.

Specific use cases will include:

- Archive
- Backup
- Business Continuity
- Cyber Resiliency
- Data Privacy
- Data Protection
- Regulatory Data Retention
- Data Security
- Disaster Recovery

The proposed architecture will not be suitable for all use cases. Specifically, high-impact, time-sensitive, high-transactional systems that may require hot-failover redundancy (or some other advanced high-performance technology) will NOT be addressed by the proposed architectural approach.

Threat Scenarios

In these examples, the recovery device within a recovery pod is the server/tape-library combination. Since the devices are hard wired with no other data path in or out except through the server, the server/tape-library combination acts as a single Zero Trust device. The requirement is for a single point of access to the storage array so that trust can be established and reset with each set of transactions.

The following items list possible attack scenarios on the Removable Media Storage Library and physical storage media, and how the architecture responds.

Malicious Encryption

- ***Adversaries want to encrypt the backed-up or archived data sets***
  Malicious encryption in this case refers to recoverable encryption – encryption with the purpose of later decryption. Tape is an append only medium. Malicious encryption by adversaries is, in fact, physically impossible with LTO/LTFS since the process

involves overwriting and replacing an existing file. All overwrites in LTFS are logical rather than physical, and a changed file can always be rolled back to a previous state.

- ***Adversaries want to delete backed up data sets***
  If somehow adversaries were to gain access to backup programs, and tried to delete the backups, this too would be impossible. All deletes on tape are logical, with standard rollback utilities that allow each tape in the system to be rolled back to a previous time.

Theft of Data

- ***Adversaries want to steal data for extortion***
  All data stored on tape should be encrypted. Encryption capabilities are embedded in standard tape-drive hardware and will render any data useless if stolen by adversaries. Varying degrees of encryption are possible. At the extreme end of the spectrum, every transaction, every file, every file system, and every application interaction CAN be encrypted. With the emphasis on recovery, all recovery assets should be encrypted and the decryption keys secured.

- ***Adversaries want to exfiltrate data – slow theft***
  Data on tape should always be encrypted. But even so, illegal exfiltration would be a concern. Therefore, the Caching Server should have alarms set to shut down the system in the case of excessive or unauthorized data exfiltration.

Malicious Disruption

- ***Adversaries insert bad files to become part of backed up data sets for later exploitation***
  Filters should be installed to examine all files before they are committed to tape. The same rollback features that disallow deletions also allow system administrations to go back in time to find clean file copies even if malicious files have been staged. The files should be restored in a way that they can be examined forensically without interacting with any live production systems.

- ***Adversaries want to damage or disable the tape library***
  In the unlikely event that a library is compromised or damaged, the tapes are severable from the library. In other words, if a library fails, tapes from the failed library can be removed and placed in a new library with zero operational or data loss.

- ***Adversaries want to corrupt the physical storage devices – tape cartridges***
  The only thing that can physically disrupt data on tapes is a procedure called a forced format, and that function can be physically disabled on a library. Tape is still viable outside of its cartridge. If physical tape destruction is truly an issue, mirrored / redundant copies should be maintained.

- ***Adversaries want to disable single-access caching server***
  If the Storage Subsystem Caching Server is compromised, systems should be configured such that the server can be wiped and restored. For more rapid system rebuild, a second server can be kept at the ready but isolated from any system activity.

After a breach, the tapes can provide the needed databases and cache material, and the secondary server can be put into operation very quickly (minutes and not hours.)

- *All connected systems are compromised*
  This is the worst-case scenario. Malicious code propagates and destroys all data on all connected systems, including cloud connections. In this case, the tapes should be configured to hold all recovery assets, the system can be restored when any hardware problems are remedied – often through wiping systems of all software and subsequently restoring a clean operating system. We refer to this strategy as WARP – Wipe And Restore Protocols.

Conclusion & Recommendations

Recovery is essential to cyber defense and resiliency. While proactive defensive planning is critical to the deterrence of cyber crime, it is widely accepted that defenses, somewhere, at some time, will be defeated. The ability to render the data stolen useless and to recover systems rapidly after an attack are both natural deterrents to cyber crime.

Cyber recovery is not a new idea; it underlies disaster recovery and business continuity response. Fortunately, catastrophic disasters are few, the downside being that our responses to them, except in technically mature organizations, are rarely tested.

Cyber attacks and cyber disasters are now commonplace. Their impact can sometimes be felt immediately, and their damage can be far more widespread than any fire or natural disaster. Old models of recovery design need to match the size and potential impact of these new threats. NIST calls out the difference between recovery from non-malicious activity and cyber attack. Every organization must now be prepared for cyber attacks.

Technologies exist, if strategically deployed, to make recovery even from the most severe cyber incidents rapid and successful. But to achieve this type of resilience it is necessary to reevaluate the definition of recovery, the needed performance requirements, and the technologies used to achieve those performance goals. This new cyber recovery architecture needs to be implemented now.

Appendix A – NIST SP 800-209 Security Guidelines for Storage Infrastructure Controls Mapping

This table outlines the recommendations in Section 4 of NIST SP 800-206 making the distinction between functional and non-functional requirements. In this case, non-functional requirements are usually procedural recommendations – things like performing regular security audits. Functional requirements can be fulfilled by a specific technical design or plan.

The table further identifies the functional requirements satisfied by the proposed architecture and gives a reference to the document to show how that recommendation is satisfied.

Security Guidelines for Storage Infrastructure NIST-SP-800-2C Cross Reference:

FR = Functional Requirements section of document
Part / ❡ = Part and Paragraph
✅ = Requirement Satisfied by Architecture

	Functional Requirement NIST SP 800-209	Capable within Architecture	Reference Section
4.1 Physical Storage Security			
PS-SS-R1 – Media security measures	No		
PS-SS-R2 – Protect all sensitive administrative equipment	No		
PS-SS-R3 – Data sanitization approach	No		
4.2 Data Protection			
4.2.1 Data Backup and Recovery			
DP-SS-R1 – Data Protection plan or policy	No		
DP-SS-R2 – Data Protection plan or policy – details	No		
DP-SS-R3 – SOPs for backup processes	No		
DP-SS-R3 – Configuration Management	No		
4.2.2 Replication and Mirroring			
DP-SS-R5 – Same level of protection for main and replica storage	Yes	✅	FR - i
DP-SS-R6 – Minimize replication trust	Yes	✅	FR - i
DP-SS-R7 – Encrypt data during transit to replication	Yes	✅	FR - i
DP-SS-R8 – Coordinate synchronous replication	N/A		

DP-SS-R9 – Delete old replicas as needed	Yes	✅	FR - i
4.2.3 Point-In-Time Copies			
DP-SS-R10 – Timing of snapshots	Yes	✅	Part 4, ❡3
DP-SS-R11 – Delete old replicas as needed	Yes	✅	Part 4, ❡3
4.2.4 Continuous Data Protection			
DP-SS-R12 – Continuous Data Protection considerations	No		
4.3 Authentication and Data Access Control	Yes	✅	Part 4, ❡7
4.4 Audit Logging	Yes	✅	Part 4, ❡7
4.5 Preparation for Data Incident Response	No		
IR-SS-R1 – Response plan for storage components	No		
IR-SS-R2 – Recovery asset immutability	Yes	✅	Part 4, ❡8
IR-SS-R3 – Verify hygiene of recovered components	Yes	✅	FR - m
4.6 Network Configuration	Yes	✅	Part 4, ❡1-❡6
4.7 Isolation			
IS-SS-R1 – Separation of Storage systems	Yes	✅	Part 4, ❡9
IS-SS-R2 – Separation of Management systems	Yes	✅	Part 4, ❡7
IS-SS-R3 – Access restriction to cyber attack recovery systems	Yes	✅	Part 4, ❡7
IS-SS-R4 – Off-site storage	Yes	✅	Part 4, ❡8
IS-SS-R5 – Independent, full baseline copy	Yes	✅	FR - d
IS-SS-R6 – Disable all unnecessary services and protocols	Yes	✅	Part 4, ❡7
IS-SS-R7 – Independence from hosts and applications	Yes	✅	Part 4, ❡7
IS-SS-R8 – Consider setting up an airgap	Yes	✅	Part 4, ❡8
IS-SS-R9 – Periodic isolation reviews	No		
IS-SS-R10 – Consider use of immutable storage	Yes	✅	Part 4, ❡8
4.8 Restoration Assurance			
RA-SS-R1 – Ensure completeness of components	No		
RA-SS-R2 – Protect dependent components	Yes	✅	FR - j
RA-SS-R3 – Availability of all relevant software and hardware	Yes	✅	FR - j
RA-SS-R4 – Component availability matches RTO	No		
RA-SS-R5 – Test recovery to match RTO	No		
RA-SS-R6 – Assess and match Recovery Point Objective	No		
RA-SS-R7 – Frequency retention requirements	No		
RA-SS-R8 – Remote replica health	No		
RA-SS-R9 – Separate data from applications	Yes	✅	FR - j
RA-SS-R10 – Document restoration plan	No		
RA-SS-R11 – Scan restoration assets for malware	Yes	✅	FR - m
RA-SS-R12 – Audit all procedures periodically	No		

4.9 Encryption			
EN-SS-R1 – Use transport layer security (TLS)	Yes	✅	Part 4, ¶ 7
EN-SS-R2 – Avoid cleartext protocols	Yes	✅	Part 4, ¶ 7
EN-SS-R3 – Encrypt storage APIs	Yes	✅	Part 4, ¶ 7
EN-SS-R4 – Encrypt administrative access	Yes	✅	Part 4, ¶ 7
EN-SS-R5 – FIPS 140-3 compliance	Yes	✅	Part 4, ¶ 7
EN-SS-R6 – Use transport layer communications	Yes	✅	Part 4, ¶ 7
EN-SS-R7 – At rest encryption	Yes	✅	FR - g
EN-SS-R8 – Encrypt between storage components	Yes	✅	FR -h
EN-SS-R9 – Key management	No		
4.10 Administrative Access	No		
4.11 Configuration Management	No		
4.12 Training	No		

About the Authors

Fred Bonner | *EchoLeaf Systems*

Fred Bonner served as an Executive Architect at IBM, has an MS in Management of Technology from the University of Maryland, and presently serves as Chief Technical Officer of EchoLeaf Systems.

Peter Guglielmino | *IBM Corporation*

Peter Guglielmino is an IBM Distinguished Engineer and has worldwide responsibility as CTO for IBM's Media and Entertainment Industry.

Dr. John Hoehn | *Life After Television*

Dr. John Hoehn is an industry strategist in digital supply chain, digital video archiving, digital cinema, and the application of business intelligence/analytics. He is a former Executive Architect for IBM, and has conducted global case study research of digital archive systems.