HARBOR LABS

SOFTWARE & NETWORKING EXPERTS

A Report on the Technical and Usability Advantages of SecurityFirst™ DataKeep™

2017 September 23

Prepared by Avi Rubin, Ph.D. and Paul D. Martin, Ph.D.

	PHONE	FAX	WEB
106 Old Court Rd. Ste. 305, Pikesville, MD 21208	443-738-4544	410-264-2406	www.harborlabs.com

Table of Contents

	PHONE	FAX	WEB
106 Old Court Rd. Ste. 305, Pikesville, MD 21208	443-738-4544	410-264-2406	www.harborlabs.com

I. Introduction

The SecurityFirst SPxCore™ is the central component of SPx products such as DataKeep. It is responsible for encrypting and splitting data to allow for secure and flexible data storage across numerous storage locations. It is comprised of a series of modules that include industry standard encryption with SecurityFirst's novel cryptographic splitting and integrated key management solution. While in a few use cases the functionality of DataKeep can be replicated with off-the-shelf components (though not the additional security of cryptographic splitting), in such cases the SecurityFirst approach is novel in many key ways, and DataKeep has several essential advantages over other solutions. Furthermore, in M:N deployments DataKeep has security advantages over existing solutions; a user must compromise M separate cloud shares as well as a secret key in order to recover encrypted data. In this report we explore the novel features of DataKeep. We also consider DataKeep configurations that can be approximated with off-the-shelf components and discuss the benefits provided by using an integrated all-in-one solution such as DataKeep.

II. Overview of SPxCore

SecurityFirst DataKeep leverages the SPxCore cryptographic splitting library in order to provide its secure and robust storage capabilities. SPxCore technology is unique in its versatility: it integrates cryptography into software RAID, creating the ability to store data in a secure and fault-tolerant fashion. Furthermore, SPxCore can use a variety of device types as its storage units, including cloud storage, flat files, logical volumes and physical volumes. Each cryptographically split portion of data created by SPxCore is called a "share".

SPxCore can be configured to provide M:N fault tolerance. This means that M valid shares must be present out of N total shares in order to decrypt and access the stored data. This is similar to RAID functionality but it is much more versatile in that the underlying devices need not be physical disks. SPxCore is also more secure since it integrates cryptography directly into the shares themselves. Thus, M:N provides more than just fault tolerance, it also provides security. That is, M:N shares in addition to an external key must be present not just to recover data, they also must be present to physically decrypt the storage volume.

SecurityFirst DataKeep builds on the SPxCore by adding numerous features to make the SPxCore easy to deploy in cloud environments. These features include integrated key-management, access control and audit logging. DataKeep is FIPS-certified and meets the requirements for deployment in a number of highly regulated environments as required by HIPAA, PCI-DSS and FIPS among other standards and regulations.

III. Security and Byzantine Fault Tolerance

DataKeep is a full-featured, integrated, secure data storage solution. In certain deployments it conveys security advantages over existing cryptographic storage systems. Leveraging the SPxCore, DataKeep can store data in a cryptographically secure and fault-tolerant fashion by splitting the data amongst encrypted shares. The SPxCore uses a modified ZFS driver to essentially create a software RAID solution on top of a number of disks or volumes (called shares). A user can create a fault tolerant system across multiple shares. This scheme is called M:N. As an example: in a 2:4 configuration there are four total data shares, yet any two can be used to reconstruct the data. DataKeep leverages technologies such as AES, Shamir Secret Splitting, and MAC and signature schemes in order to cryptographically secure data-at-rest.

Shamir's secret sharing, or threshold scheme, is a method whereby a trusted party computes some number of secret shares S_i from an initial secret S. The number of secret shares is greater than or equal to 1 and less than or equal to the predefined threshold T. The trusted party distributes the secret shares S_i to some number of parties P_i. If the parties, collaborating in a group, have secret shares greater than or equal to the threshold T then they recover the initial secret S. Else, groups with secret shares of size $T-1$ or fewer will not recover the initial secret.

Shamir secret sharing is used to split a KEK (key-encrypting key) into shards. These shards are used to encrypt a share encryption key which has in turn been encrypted with a policy key; the share data is encrypted with the share encryption key. The encrypted share encryption key and shard are stored in the header of a share. In DataKeep the policy key is stored in a keystore called the Policy, Provisioning and Management server (PPM). To decrypt the data, both the KEK and the policy key are required. The KEK may only be reconstructed and used to decrypt the session keys with M of N headers present.

Data stored in shares is encrypted using AES in either CTR, CBC or GCM mode with a 128, 192 or 256-bit key. The encrypted data is then optionally authenticated using either an HMAC with SHA1, SHA256, SHA384, SHA512 or GMAC (for secure modes), a signature algorithm such as DSA or ECDSA or AES-GCM.

The RAID-like fault tolerance designed into DataKeep is a natural fit for any dispersed storage application, as robustness against geography-based outages is a basic requirement in ensuring uptime.

DataKeep also allows a user to define shares to reside on cloud storage locations, thus creating a fault-tolerant, secure cloud storage system. Furthermore, because DataKeep is secure in the face of n-m malicious cloud providers, the system is Byzantine fault tolerant. To my knowledge, no other security solutions today provide this type of Byzantine fault tolerance backed by cryptography.

This M:N configuration also has security advantages over existing cryptographic storage systems. By leveraging this M:N functionality across a cloud storage backend, an organization can split its shares across different geographic locations while simultaneously improving the security of the data-at-rest by increasing the number of compromises that must occur before data can be feasibly reconstructed. An attacker that can recover a master key still must compromise several cloud storage providers in order to reconstruct the data. This is tangibly more difficult than compromising a single cloud storage provider as it increases the complexity of the attack and the likelihood that the attacker will be discovered. M:N configurations thus improve the overall security posture of the underlying system compared to standard encrypted cloud storage.

Note that this fault tolerance and enhanced security is only gleaned in the case that DataKeep is deployed in an M:N configuration. If, instead, DataKeep were deployed in a 1:1 configuration, there would be no fault tolerance or cryptographic advantages over other solutions. The above described infrastructure and header computation remains in place, but the KEK is simply stored in the share header and the system security is reduced to that of the locally stored policy key. However, even without the key splitting there would still be numerous other advantages outside of fault tolerance and Security. All of the advantages described in the following section apply to DataKeep regardless of whether it is deployed in a 1:1 or M:N configuration.

IV. Key Management

DataKeep contains a keyserver called the Policy, Provisioning and Management (PPM) server to facilitate secure key storage. In addition, DataKeep also supports other external keystores. This allows cloud-hosted DataKeep VMs to store cryptographic keys on a system outside of the control of the cloud provider.

The DataKeep PPM provides the ability to protect cryptographic keying material in order to prevent keys from being stored in plain-text, where they can be more easily compromised. Without a secure keystore, an attacker that has subverted a PPM server may be able to simply read and exfiltrate encryption keys. A keystore typically 'wraps' individual keys with a master key before storing them on-disk. This entails encrypting each key with a master Key-Encrypting Key (KEK) and storing the corresponding ciphertext, thereby ensuring that only through a brute-force attack or through theft of the KEK could an attacker retrieve these keys.

A more secure keystore might make use of a HSM (Hardware Security Module). An HSM is a hardware appliance that specializes in secure storage of cryptographic keys and uses standard interface protocols such as PKCS#11 or KMIP. An HSM will normally generate the KEK securely within hardware and prevent any software requests to read-out the key. Instead, the user can use commands to instruct the HSM to use the KEK for cryptographic operations that take place only inside the HSM.

In this way, these keys can only be decrypted by the specific HSM that encrypted them, and an attacker cannot exfiltrate the KEK. To provide further protection, HSMs are often tamper-resistant. For example, if an attacker tries to physically tamper with the HSM, it may wipe all hardware-resident master keys. This prevents the attacker from ever reading out the keys, at the cost of no longer being able to use any keys that were encrypted within the HSM.

DataKeep supports the use of an HSM (PKCS#11 or KMIP compatible) key manager to secure the policy keys. The PPM server provides a centralized, networked service that allows authorized clients to retrieve and use cryptographic keys. By consolidating key generation, storage, and access control to a single service, the complexity of key management can be greatly reduced with DataKeep. This is especially true in complex network architectures.

For example, without a keyserver like PPM, an administrator may have to distribute the same key to any client machine that requires it. This dissipates the responsibility of securing the key to each client, who must take care to protect the key when it is stored at rest. This exposes a much greater attack surface, as an attacker that subverts any of the clients may be able to retrieve the key. If a client machine is misconfigured or insecure, it may leak the keys used by all other client machines.

With a keyserver, as supported by DataKeep, each client can authenticate to the server and request the key on a case-by-case basis. Under this model, the client need not concern itself with protecting the key at-rest. Instead, the client machine only needs to remove it from memory after it has been used for cryptographic operations. If an attacker subverts the machine, the key material will not be available on-disk. More importantly, a keyserver can be used to keep keys within the realm of control of an organization, even when an application leveraging the keys is located outside of the organization, for example on a cloud provider.

Additionally, by using a single 'source of truth' for cryptographic keying material, the keyserver can better enforce policies such as access control and key revocation. For example, if it has been determined that a client machine has become infected by malware, an administrator can easily revoke the client's credentials and prevent them from accessing any keys in the future. Without a key server, an infected client would have to store these keys themselves, and thus they could be stolen. Likewise, if a key itself has been found to be compromised, an administrator can prevent any future client queries for this key and can initiate a re-keying process that requires no re-distribution of keys to clients. They must simply replace the key on the server and ensure that any application data is re-encrypted to use this new key.

Given the paramount importance of securely managing and storing keys, the fact that DataKeep contains built-in key management but also supports external key servers is one of the major advantages of the system. In fact, in supporting the use of external key servers, DataKeep allows data to be stored and encrypted in the cloud without requiring the associated secret keys to ever be stored off-site.

V. Access Control and Audit

Layered atop this secure, fault-tolerant cloud storage is a policy based access control system. The access control system allows for fine-grained role-based access based on the least-privilege principle. In a role-based access control system (RBAC), users are assigned to particular roles, and roles are given privileges such as read or write on a per-file or per-directory basis. In RBAC systems, access control is affirmative. That is, accesses are denied by default unless there is a provision in the access control policy to allow for a particular access.

To further strengthen this feature, DataKeep allows for access controls to prevent even system administrators from reading unencrypted data. In many situations, the specific data being stored in the system are not relevant to a system administrator trying to manage the system. In DataKeep, privileged users can manage the system by setting system-wide policies without ever being able to see the underlying data. This is particularly useful in deployments where data privacy is of paramount importance, such as in healthcare.

Most large organizations typically centralize their user and policy management in a directory database such as Active Directory or LDAP. DataKeep can integrate these services in order to allow administrators to easily import and manage users. This feature allows DataKeep to be much more easily integrated into an enterprise environment than many other solutions, which would be much more complex to integrate, if it were even possible at all.

Additionally, DataKeep supports detailed audit logging - a critical feature for many deployment scenarios. This feature can be used to create a record of whenever a user accesses, creates or modifies a file, or whenever such actions are denied by the RBAC system. This audit log can then later be reviewed in order to detect misuse of the system or anomalous material.

There are numerous scenarios in which this type of audit logging is mandated by law. For example, HIPAA requires healthcare providers to keep detailed logs of all accesses in a system; these logs may later be analyzed if some type of misuse occurs. In these types of scenarios, it is imperative that a product keep detailed audit logs. Many solutions on the market cannot be configured to do so and thus cannot be deployed in such environments; DataKeep has the distinctive ability to fulfill the audit log requirements in these situations.

Even when audit logs are not legally required, keeping detailed records of system accesses is a fantastic way to detect abuse, misuse or compromise. Audit logs become even more powerful when combined with other sources of usage information such as network traffic logs and IDS events. To this end, DataKeep audit logging also integrates with SIEM (Security Information and Event Management) products for automated correlation of anomalous activities with other network and system-wide indicators and for automated threat detection.

SIEM products are commonly used by large organizations to correlate security events and potentially detect attacks as they happen rather than significantly after the fact. By having an encrypted cloud storage system integrate with a SIEM, administrators can detect and respond to threats much more quickly than would otherwise be possible.

VI. Ease-of-Deployment

While it is generally possible to cobble together different solutions with some shared aspects that may share some aspects with DataKeep, there are numerous pitfalls involved – in doing so that only a system administrator with highly developed security skills can hope navigate the difficult decisions and obstacles inherent in creating an alternative solution.

Which cryptographic algorithm is most secure? Should a MAC be used? How should keys be stored? How can RBAC policies be integrated with Active Directory? Such questions require highly specialized knowledge to answer correctly, and necessitate nuanced consideration of specific details. Even the experts get cryptographic configuration details wrong at times.

In a commercial, integrated solution such as DataKeep, these issues have already been carefully considered, rigorously tested, and successfully solved. A user can ultimately manage complex security through an easy to use web interface or REST API. Furthermore, even without significant training a user can be assured that he or she will arrive at a secure deployment. When building a system from pre-existing components, many pieces may be overlooked or neglected, particularly audit and secure key management, but in an easy-to-use solution such as DataKeep, these components are front-and-center. Features are directly integrated into the core product and are thus easy to configure and, most importantly, use correctly.

DataKeep is able to shield the user from the potentially detrimental effects of complexity, while concurrently boosting the user's ability to configure secure solutions, by providing a well-designed web interface and easy-to-configure modules for all aspects of the system, including RBAC policy and key-management. Thus, the value of even a 1:1 deployment of DataKeep is the level of engineering that was invested in producing a secure, yet elegantly simple solution for users. This is not a result that could be easily replicated without a significant engineering undertaking, and it is the most important contribution of the entire system.

VII. Conclusions

DataKeep is an innovative, secure data storage solution that is easy to deploy. It can be configured to provide high security with minimal expertise; allowing administrators to avoid pitfalls that are easy to fall into without deep cryptographic knowledge. DataKeep has many advanced features that are not commonly found together in an all-in-one solution include RBAC, Active Directory integration, secure audit and SIEM integration. It can be configured to provide Byzantine fault tolerance, a completely novel feature. Finally, DataKeep has been officially certified for deployment in Red Hat 6.2+ and 7.2+, Microsoft Server 2008 R2+, and VMware environments.

	PHONE	FAX	WEB
106 Old Court Rd. Ste. 305, Pikesville, MD 21208	443-738-4544	410-264-2406	www.harborlabs.com