Deploy Next-Generation Cloud Applications on Apache Cassandra with Datos IO RecoverX on Cisco UCS and Cisco ACI

Executive Summary

The Cisco® Application Centric Infrastructure (Cisco ACI™) solution provides an elastic, controller-based software-defined networking (SDN) fabric. With Cisco ACI, you can provision, manage, expand, and troubleshoot any workload anywhere using a single point of management. Through the single point of management provided by the Cisco Application Policy Infrastructure Controller (APIC), troubleshooting is a simplified top-down operation, with APIs that enable automation and integration with many systems, including numerous third-party systems.

Integration with Datos IO RecoverX provides transparent protection for your agile Mode 2 applications, such as real-time analytics, Internet of Things (IoT), and machine learning, in both your private, on-premises infrastructure and your cloud-based systems. Cisco ACI solution-specific features enhance Datos IO RecoverX. In particular, Cisco ACI enables zero-touch addition and removal of Cassandra nodes in a cluster backed by Datos RecoverX, providing an elastic environment that you can rapidly change to meet your needs.

Cisco ACI Overview

The Cisco ACI solution reduces total cost of ownership (TCO), automates IT tasks, and accelerates data center application deployment. It provides these benefits by using a business-relevant SDN policy model across networks, servers, storage solutions, security solutions, and services. Cisco ACI in combination with Cisco Nexus® 9000 Series Switches provides superior performance, deep application insight, and a unified solution to automate the data center from end to end, while keeping the needs of the application foremost.

Datos IO RecoverX Overview

Business growth, digital transformation, and business innovation are propelling enterprises to adopt next-generation, high-value, data-centric applications. Such applications include content-management systems, real-time analytics, operational intelligence, artificial intelligence, machine learning, and IoT processes. To support the data requirements of these high-volume, high-ingestion-rate, next-generation applications, enterprises are rapidly adopting massively scalable and nonrelational databases such as Apache Cassandra, MongoDB, Apache HBase, and Amazon DynamoDB.

These next-generation cloud applications and distributed databases are always on, distributed, and geographically replicated, leading to new data protection requirements including:

- Online application-consistent backup operations
- Near-zero recovery point objective (RPO) and recovery time objective (RTO) metrics
- Highly specific recovery capabilities for test and development and nonrecovery use cases
- High-order storage efficiency for backup data in cloud storage
- Holistic data protection across relational and nonrelational databases

Traditional data-protection products are not designed for the hyperscale, distributed nature of these next-generation applications and cannot support these requirements, creating a critical data-protection gap.

As enterprises migrate traditional applications to the cloud and deploy next-generation applications natively in the cloud, Datos IO can help. Datos IO delivers protection, mobility, and monetization solutions for hyperscale, distributed applications, enabling organizations to harness the full power of their data in a cloud-first world. Datos IO RecoverX is an industry-first, scale-out application-centric data protection software-only product built specifically for hyperscale, highly distributed, next-generation cloud applications. Enterprises can now use the continuous data-protection capabilities of Datos IO RecoverX to scale business-critical applications with confidence that their data can be recovered and that a high degree of application uptime can be maintained.

Datos IO RecoverX Features

Datos IO RecoverX is purpose-built to protect hyperscale, highly distributed, next-generation applications. RecoverX allows organizations to protect their data at any level of detail and at any point in time (flexible RPOs). It reduces downtime, recovering data in minutes, not hours. It saves organizations up to 70 percent in secondary-storage costs and increases the productivity of application owners and DevOps teams (Figure 1).

The main features of the product include:

- Cloud-first, scale-out software
- Scalable versioning for application-consistent point-in-time backups
- Single-click recovery for operational recovery, compliance, and test and development use cases
- Industry-first semantic deduplication for outstanding storage efficiency

Figure 1. Datos IO RecoverX: Application-Centric Data Protection for the Cloud

Datos IO RecoverX is founded on Consistent Orchestrated Distributed Recovery (CODR), Datos IO's cloud-first, scale-out data management architecture, enabling customers to harness the cloud for next-generation data protection and management. With 16 patents approved or pending, CODR uses elastic computing services that can automatically scale with the load. It eliminates dependency on media servers, and it transfers data in parallel to and from file-based and object-based secondary storage for multiple workloads, including data protection and testing and development.

Cisco ACI and Datos RecoverX: Better Together

With the combined power of Cisco ACI, Datos RecoverX, and Apache Cassandra, applications can expect superior performance, deep application insight, and a modern backup strategy.

Cisco ACI application profiles model the pieces of an application into endpoint groups (EPGs), or tiers. The profiles use contracts to specify which pieces can talk to each other, and they use filters to specify what they can talk about (ports, protocols, etc.). After an application profile is modeled, new endpoints can be added without making any changes to the existing policy. After the Cisco ACI fabric understands the intended policy, more nodes can be added without the need for any user intervention.

Cisco ACI: Zero-Touch Addition and Removal of Cassandra Nodes

The addition and removal of Apache Cassandra nodes to a cluster demonstrate the power of Cisco ACI application profiles. The next section discusses the profile used to model the Datos RecoverX application and Cassandra nodes.

In an existing environment, you frequently may need to add more Cassandra nodes to a cluster to accommodate growth in the database, increase resiliency, or add computing power. In the past, each node added required manual configuration of the network infrastructure. This process could take a long time, because the needs of application teams cannot readily be translated into traditional network objects such as VLANs, subnets, IP addresses, quality-of-service (QoS) policies, access control lists (ACLs), etc.

With Cisco ACI, however, adding a new node is zero-touch operation (aside from the physical installation). Cisco ACI already has the application profile and components needed to identify the traffic from the new node when it reaches the Cisco ACI fabric, and it can enforce the policy at line rate in hardware immediately. You do not need to notify Cisco ACI of the new nodes.

When Datos RecoverX with Apache Cassandra is run in a Cisco ACI environment, the cluster can elastically expand and contract without the need to manage the details of the network infrastructure underneath. Modern, application-centric policy already is in control and knows how to adapt.

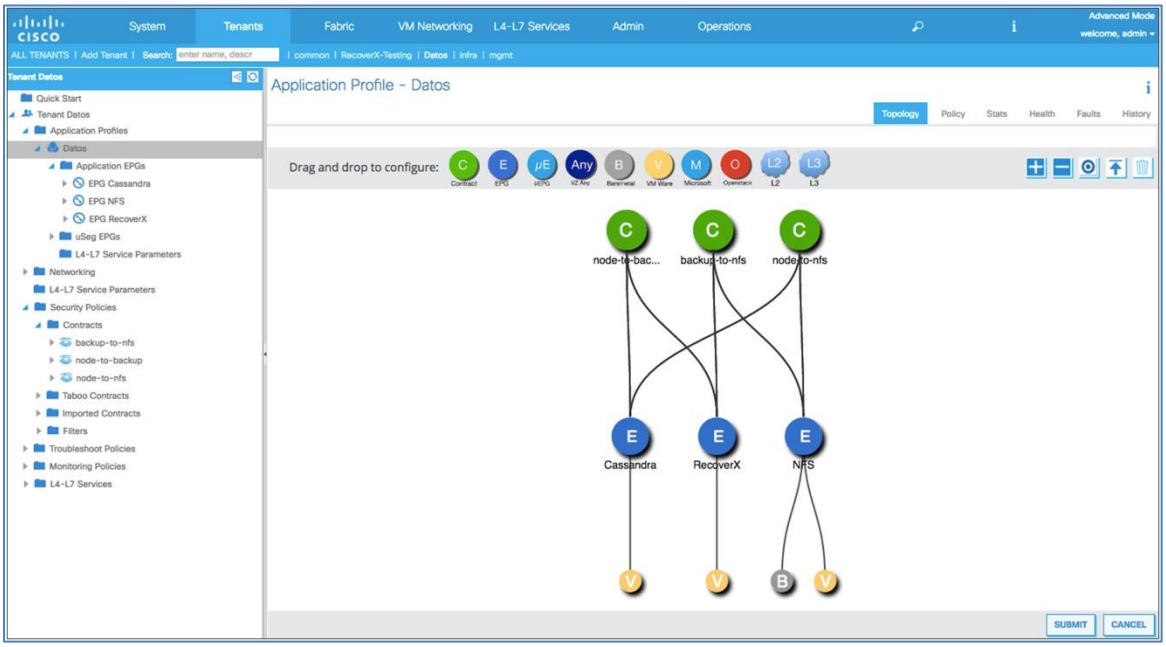
Cisco ACI Application Profile for Datos RecoverX and Apache Cassandra

In the lab validation environment discussed in this document, an application profile was created to model the backup and database applications that govern the flow of communication between components, permitting only the traffic required. Cisco ACI operates based on a whitelist model, in which no traffic is permitted until a policy is specifically implemented, providing an environment with inherently superior security than a classical Ethernet environment.

The Cisco ACI application profile defines an EPG for the Cassandra nodes, an EPG for the Datos RecoverX nodes, and an EPG for the Network File System (NFS) server used for backup. Contracts are enforced to help ensure that different EPGs can talk to each other and to restrict traffic to only the ports required for communication.

As shown in Figure 2, the node-to-backup contract governs communication between the Cassandra database nodes and the RecoverX service, the backup-to-nfs contract governs communication between the RecoverX service and the NFS servers, and the node-to-nfs contract governs communication between the Cassandra database nodes and the NFS servers.

Figure 2. Datos IO RecoverX Application Profile in Cisco ACI

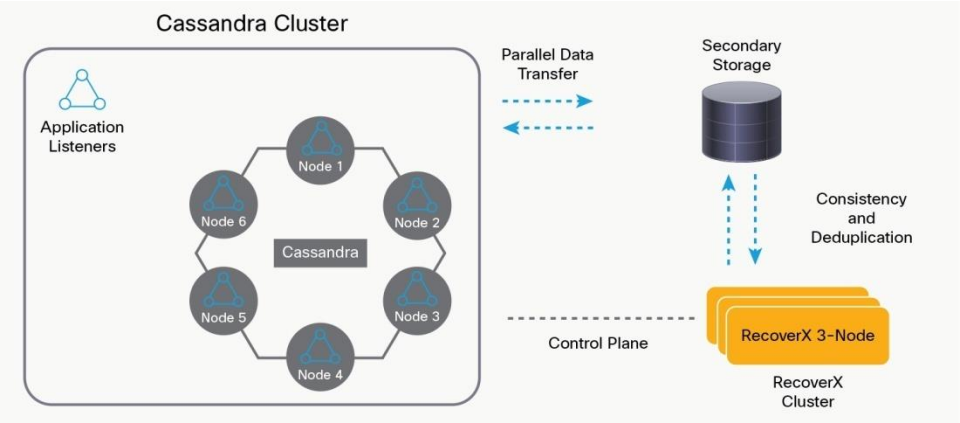


Note that any or all of the endpoints in the EPGs can be on any hypervisor, bare-metal, or container service. Cisco ACI functions the same way regardless of where the application components are actually running. Devices running on different hypervisors and bare-metal systems are not penalized, classified, or treated differently based on their host system. A Cassandra node is a Cassandra node, regardless of whether it is in a virtual machine or running on bare metal.

Setting Up Datos RecoverX in a Cisco ACI Environment

The Datos IO RecoverX software is easy to install and configure. After you prepare the deployment environment that will host RecoverX, the installation of RecoverX requires only a few steps. After installation is complete, a user can connect to the RecoverX console and immediately begin to configure and manage data sources, versioning policies, version storage, and recovery operations. The example in Figure 3 shows Datos IO RecoverX when it is deployed to protect a database cluster.

Figure 3. Datos IO RecoverX Deployed to Protect a Database Cluster



The deployment contains the following components:

- Data source (database cluster): The data source that the customer wants RecoverX to protect: for example, a cluster of Cassandra or MongoDB databases
- Secondary (backup) storage: The device on which the versions of the data source are stored
- Datos IO RecoverX: Software that contains the CODR engine, which is responsible for all operations related to versioning and recovery as well as for any internal and background maintenance operations
- Datos IO application listeners: Lightweight listeners that communicate with the data sources and that are automatically deployed, operated, and managed by Datos IO RecoverX

RecoverX is deployed in three-node cluster configuration.

To deploy RecoverX, use the following steps:

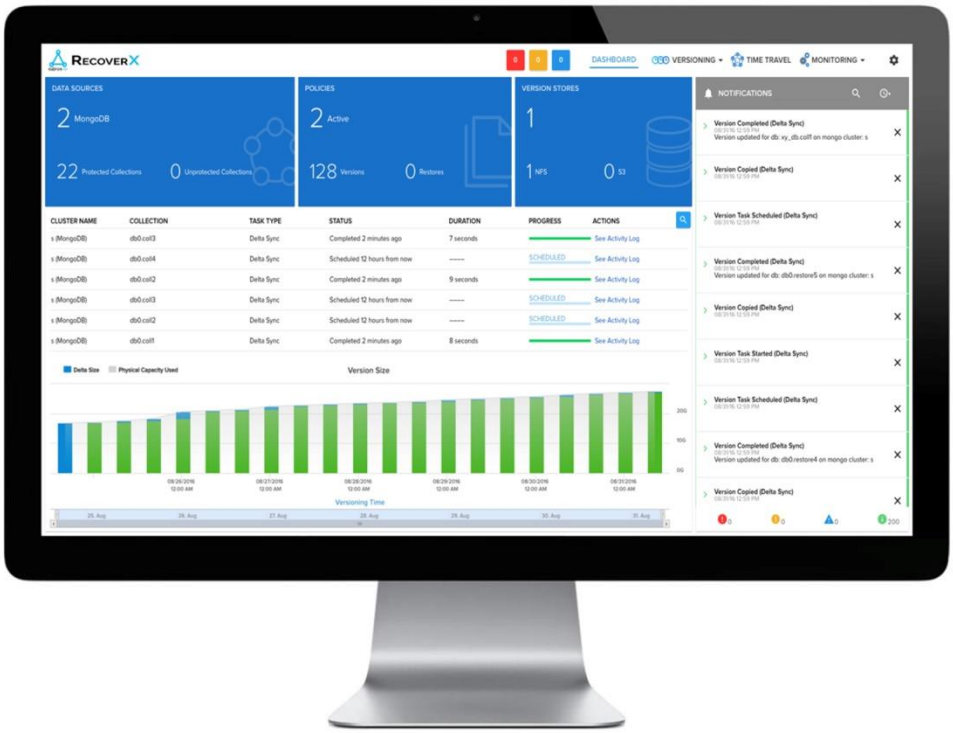
1. Prepare the data source (database cluster).
    a. Create a Datos IO user on each data source node.
    b. Configure Secure Shell (SSH) and enable read and write permissions on certain directories.
2. Prepare the server for RecoverX.
    a. Create a virtual machine and Datos IO user on each node.
    b. Enable **sudo** privileges for certain commands and configure the **nproc** and **nofiles** parameters.
3. Configure NFS storage.
    a. Create an NFS share file in which the backup data will be stored.
    b. Hard-mount the NFS file share on all data source and RecoverX nodes.
4. Configure the network.
    a. Data path: Set up connectivity from the data source and RecoverX nodes to the storage.
    b. Control path: Set up SSH connectivity from RecoverX to the data source nodes.
    c. Firewall settings: Open certain network ports on the RecoverX cluster nodes.

5. Install RecoverX.
    a. Copy the tarball file to one of the RecoverX nodes, expand (untar) the file, and issue the **install** command.

For an in-depth description of the installation process, click here.

After RecoverX is installed, log in to the GUI from https://<IP_address>:9090/#/dashboard (Figure 4).

Figure 4. Datos IO RecoverX GUI



Detailed Installation Steps

Follow these steps to create the environment and install Datos IO RecoverX:

1. Create the Cisco ACI fabric and set up networking

    For help setting up a Cisco ACI network fabric, please refer to this document: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/2-x/getting-started/b_Getting_Started_Guide_Rel_2_x.html.

2. Set up hosts for the virtual machine infrastructure.

    For help setting up a VMware infrastructure, please refer to VMware's guidelines for best practices: https://pubs.vmware.com/vsphere-60/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-60-installation-setup-guide.pdf.

3. Set up the host for NFS.
    a. For the setup discussed in this document, use a bare-metal server for NFS.
    b. Install CentOS 6.8 or later as the operating system.

c. Set up networking (IP addresses, Domain Name System [DNS], host file, and routing).
    i. For a standard network configuration, either use internal DNS by editing the /etc/resolv.conf file to resolve all the host names for the environment, or edit the /etc/hosts file.
    ii. Shut down the internal firewall. You can do this by entering the command **#service iptables stop.** To disable the firewall from startup, enter the command **#chkconfig iptables off/**.
d. Set up Network Time Protocol (NTP).
    i. Run the command **#yum install ntp ntpdate ntp-doc**.
    ii. If you want to use an internal NTP server, add the server by modifying the /etc/ntp.conf file.
    iii. Turn on the service by entering **#chkconfig ntpd on**. (This command also sets the service to start during bootup).
    iv. Synchronizer the clock immediately by running **#ntpdate [server name or ip]**.
e. Install NFS. (Use NFS Version 3, not Version 4. Versiion 4 is required for compatibility reasons).

4. Create three virtual machines for Cassandra nodes.
    a. Specify these settings (minimum requirements): CPU = 4, RAM = 16 GB, and disk = 256 GB.
    b. Install CentOS 6.8 or later.
    c. Set up the network interfaces (IP, DNS, host file, and routing).
        i. For a standard network configuration, either use internal DNS by editing the /etc/resolv.conf file to resolve all the host names for the environment, or edit the /etc/hosts file.
        ii. Shut down the internal firewall. You can do this by entering the command **#service iptables stop**. To disable the firewall from startup, use the command **#chkconfig iptables off**.
    d. Install Java.
        i. Run **#yum install java**.
    e. Set up NTP.
        i. Run the command **#yum install ntp ntpdate ntp-doc**.
        ii. If you want to use an internal NTP server, add the server by modifying the /etc/ntp.conf file.
        iii. Turn on service by entering **#chkconfig ntpd on**. (This command also sets the service to start during bootup).
        iv. Synchronize the clock immediately by running **#ntpdate [server name or ip]**.
    f. Mount the NFS file share.
        i. Install the NFS mount binaries by running the command **#yum install nfs_utils nfs-utils-lib**.
        ii. Create a directory for the share: for example, enter **#mkdir -p /mnt/nfs/[fileshare name]**.
        iii. Mount the NFS shared drive by entering the command **mount [ip addr of NFS server]:/[path of NFS share] /mnt/nfs/[fileshare name]**.
        iv. Check the setup by running **#df –h**.
        v. Run **#mount -o auto, hard, actimeo=0, lookupcache=none, noac, nfsvers=3 [ip addr or hostname of NFS server]:/[path of NFS share] /mnt/nfs/[fileshare name]**.

            Note: You need to specify **nfsvers=3** only if you are running a newer version of NFS such as Version 4. This setting will limit the setup to NFS Version 3.

g. Add users as described in the RecoverX installation document.

   i. Run the command **#useradd -g cassandra -m datos_db_user**.

   ii. Run the command **#passwd datos_db_user**.

   iii. Run the command **#chown -R nobody:nobody /mnt/nfs/[fileshare name]**.

   iv. Run the command **#chmod -R 777 /mnt/nfs/[fileshare name]**.

h. Configure the maximum number of SSH sessions as described in the RecoverX installation document.

5. Create one virtual machine for Datos IO RecoverX.

   a. Specify these settings (minimum requirements): CPU = 8, RAM = 32 GB, and disk = 140 GB.

   b. Install CentOS 6.8 or later.

   c. Set up network interfaces (IP, DNS, host file, and routing).

      i. For a standard network configuration, either use internal DNS by editing the /etc/resolv.conf file to resolve all host names for the environment, or edit the /etc/hosts file.

      ii. Shut down the internal firewall. To do this, enter the command **#service iptables stop**. To disable the firewall from startup, enter the command **#chkconfig iptables off**.

   d. Set up NTP.

      i. Run the command **#yum install ntp ntpdate ntp-doc**.

      ii. If you want to use an internal NTP server, add the server by modifying the /etc/ntp.conf file.

      iii. Turn on service by entering **#chkconfig ntpd on**. (This command also sets the service to start during bootup.)

      iv. Synchronize the clock immediately by running **#ntpdate [server name or ip]**.

   e. Mount the NFS file share.

      i. Install the NFS mount binaries by running the command **#yum install nfs_utils nfs-utils-lib**.

      ii. Create a directory for the share: for example, enter **#mkdir -p /mnt/nfs/[fileshare name]**.

      iii. Mount the NFS shared drive by using this command **mount [ip addr of NFS server]:/[path of NFS share] /mnt/nfs/[fileshare name]**.

      iv. Check the setup by running **#df –h**.

      v. Run **#mount -o auto, hard, actimeo=0, lookupcache=none, noac, nfsvers=3 [ip addr or hostname of NFS server]:/[path of NFS share] /mnt/nfs/[fileshare name]**.

      Note: You need to specify **nfsvers=3** only if you are running a newer version of NFS such as Version 4. This setting will limit the setup to Version 3.

6. Install DataStax Enterprise (DSE) 4.8.5 on the three Cassandra nodes.

   a) Download DSE 4.8.5 from DataStax (be sure to register to download for free): https://academy.datastax.com/downloads/download-previous-versions-dl-enterprise.

   b) Follow the instructions located at https://docs.datastax.com/en/datastax_enterprise/4.8/datastax_enterprise/install/installNoSudoOrMac.html.

   Note: For the test installation described in this document, Advanced installation option was selected, and then the default settings were selected. This process was used because the Simple installation option failed for the test installation.

7. At this point, the Cassandra cluster nodes should have an NFS share mounted on each node, and the Datos RecoverX server should also have an NFS share mounted. Network connectivity should exist between the cluster, the NFS storage, and the RecoverX server. To verify, make sure that **ping** is successful between each component host. Ping by host name to simultaneously test name resolution.
8. Install the Datos.IO RecoverX application as described earlier in the section "Setting Up Datos RecoverX in a Cisco ACI Environment."

Cisco ACI and Datos RecoverX with Apache Cassandra Lab Environment

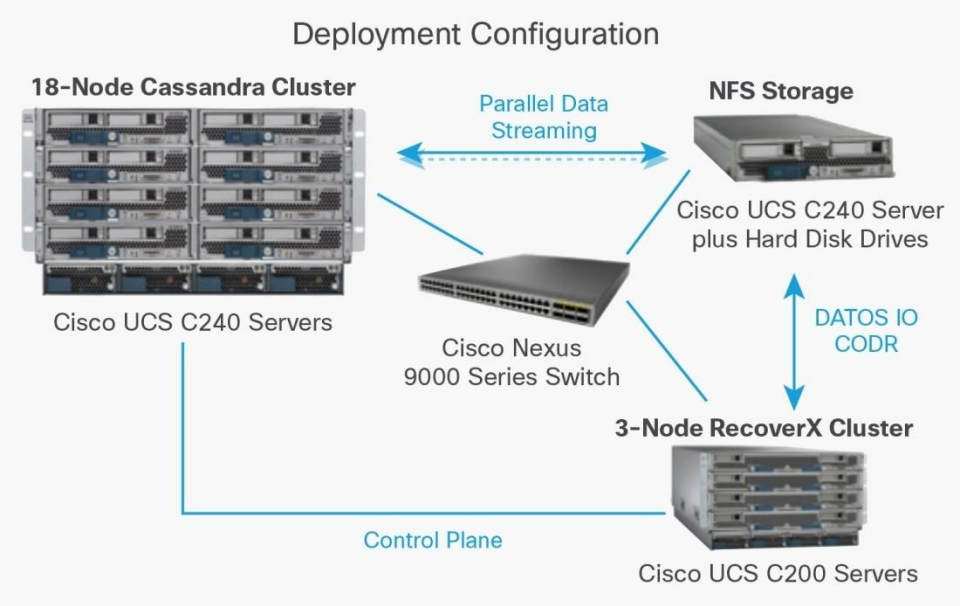
Datos IO RecoverX software, Cisco Unified Computing System™ (Cisco UCS®) servers, and Cisco Nexus 9000 Series Switches running Cisco ACI with an APIC cluster were used to validate the power of a combined solution. A large-scale Cassandra environment was created using the following:

- Cassandra database: 18 nodes deployed on Cisco UCS C240 bare-metal rack servers
- Datos IO RecoverX: 3 nodes deployed on Cisco UCS C200 bare-metal rack servers
- Secondary storage: NFS server using Cisco UCS C240 with large-capacity hard drives

The Cassandra database cluster was loaded with approximately 4 TB of data in 12-column families with a replication factor of 3. RecoverX was then used to back up the entire database cluster and store the backup data in NFS storage.

Figure 5 shows the configuration.

Figure 5. Lab Environment



Using CODR architecture that removes media-server bottlenecks, RecoverX was able to complete the initial backup of the entire cluster within 1.6 hours. This result is equivalent to about 60 TB per day of backup performance. And because of the solution's industry-first semantic deduplication capability, only 1.4 TB of storage capacity was used on the NFS server.

Conclusion

As more next-generation applications use highly distributed nonrelational databases such as Apache Cassandra and MongoDB, catering to the needs of the application becomes increasingly important from the perspective of the infrastructure. Data protection continues to be fundamental, so when you combine next-generation solutions such as Cisco ACI, Datos IO RecoverX, and Apache Cassandra, you achieve benefits immediately. RecoverX provides a modern approach to backup, Cassandra is increasingly used as the database model of choice, and Cisco ACI provides a transparent, zero-touch approach to infrastructure and truly bridges the gap to provide an application-centric approach to the modern data center.

For More Information

http://www.cisco.com/go/aci
http://datos.io/solutions/cassandra-backup-and-recovery/

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-738617-00 03/17