January, 2022 OpenNTF Webinar
Backup your Domino Server - New Options in V12
AGENDA

• Welcome – Howard Greenberg and Graham Acres
• Presentation – Daniel Nashed, Nash!Com
• Q and A - All
THANKS TO THE OPENNTF SPONSORS

• HCL made a contribution to help our organization
  • Funds these webinars!
  • Contests like Hackathons
  • Running the organization

• Prominic donates all IT related services
  • Cloud Hosting for OpenNTF
  • Infrastructure management for HCL Domino and Atlassian Servers
  • System Administration for day-to-day operation
THIS IS OUR COMMUNITY

• Join us and get involved!
• We are all volunteers
• No effort is too small
• If your idea is bigger than you can do on your own, we can connect you to a team to work on it
• Test or help or modify an existing project
• Write guides or documentation
• Add reviews on projects / stars on Snippets
UPCOMING EVENTS

• Engage 2022 – Bruges, Belgium – March 22-23, 2022
  • https://engage.ug/
NEXT WEBINAR - TBD

• February 17, 2022
ASKING QUESTIONS

- First Question – Will this be recorded?
  - Yes, view on YouTube!!!
    - https://www.youtube.com/user/OpenNTF
- Use the Questions Pane in GoToWebinar
- We will get to your questions at the end of the webinar
- The speakers will respond to your questions verbally
  - (not in the Questions pane)
- Please keep all questions related to the topics that our speakers are discussing!!!
- Unrelated Question => post at:
  - http://openntf.slack.com/
BACKUP YOUR DOMINO SERVER - NEW OPTIONS IN V12

Daniel Nashed
Domino 12.0.1 Backup & Restore

Daniel Nashed
HCL Lifetime Ambassador
Introduction & Functionality
Domino Backup Requirements

- Domino NSF files need to be backup-ed on-line
  - Open file backup is not an option and leads to inconsistent databases!
  - You either need a Domino aware backup software
  - Or shutdown your Domino server for backup!

- VSS (Volume Shadow Copy) or file-system snapshots on Linux alone are not supported
  - Domino Backup integration needed to support Snapshot backup

- Customers ask for archive style transaction logging to allow point in time restores

- New requirements like backup Docker containers and also new backup vendors
  - For example Veeam leveraging snapshot backup
Domino V12 Design Goal

- Full featured Domino Backup integration to bridge between Domino and backup solutions
- Allow integration with any type of backup vendor via custom scripts
- Allow Domino to integrate with customer’s favorite backup solution
- Support snapshot backup applications
- Allows broad functionality on the Domino backup/restore side
- Notes/Domino style UI in combination with favorite backup solution
Main Functionality

• Flexible restore options
  − Bring database online, point in time recovery,
  − Disable replication, change replica ID, change title, disable all agents, etc.

• Command-Line/REST API, etc. integration for backup tools with file and snapshot backup

• Support for **Win64** and **Linux64** only
  − AIX and OS400 have strong IBM vendor support already

• Recover **documents** and **folders** into original database

• Integrated DAOS restore missing NLOs leveraging an existing file-backup
  − Focus is NSF & Translog backup. DAOS is a single file backup handled by any backup application
Technical Background

• **Problem**
  - Domino is constantly accessing files on disk; thus, file copy operations are causing inconsistent backups.

• **Solution**
  - **Domino Backup API** brings database into consistent state
  - Now NSF/NTFs can be backed up on file-level
  - Write access is still available to end users.

• Changes (aka deltas) are captured and **must** be applied to the backup to bring the database into consistent state!
  - Delta files are merged during backup in case of file backup operations
  - Or on restore in case of other operations
  - The Domino backup & restore takes care of tracking delta files
Backup and Restore Application Components

- **backup** – Backup server task invoked via program document

- **restore** – Restore server task either
  - a.) invoked on command line or
  - b.) running permanently monitoring the restore job database via -g option

- **dominobackup.nsf**
  - Configuration
  - Database Inventory and Backup Logs
  - Restore interface and restore jobs
dominobackup.nsf

- Configuration
  - Per server or global configuration per platform
  - Global settings

- Per Database Backup log repository
  - Showing all backups per database and base for restore operations

- Per Backup log for NSF files and Translog

- Restore requests
  - Derived from per database backup, when you select a restore
Backup Concept: Full Backup

- Full backup only
  - Allows to recover data at the point in time when the backup was taken

- With circular Transaction logging is enabled in Domino
  - Limited point in time recovery if transaction logs are still on disk

- Typical setup:
  - Daily Full backup
  - Backup job running outside of business hours
Backup Scenario: Disk / File Share

• Configured out of the box
  − Just needs a “disk” with sufficient space to backup databases
  − Very simple to setup

• Changes during backup in the databases are automatically merged into the database backup!
  − Consistent backup without the need to use the restore
  − Databases can be just copied back – Don’t overwrite existing databases on OS level when the server is up!!

• Any type of file-share
  − Recommended: Storage supporting compression and de-duplication
    ▪ Cohesity → Backup share into backup repository (called “View”)
    ▪ e.g TrueNAS → https://www.truenas.com/ - leverages ZFS storage like many others on the market
    ▪ Other deduplicating storage like NetApp etc..
Quick Start Backup

1. **load backup** → creates new dominobackup.nsf

2. Review configuration
   - One pre-defined file copy configuration per platform
   - Already enabled
   - Check if the target location works in your environment
   - Server needs write permissions to create the directory!

3. **load backup** → to start your first backup
Demo: Backup Jumpstart

Run Domino backup with File Integration
Restore Operations

1. Find the database to restore
2. Create a “restore job” using the action button
3. Specify the restore time and restore options
   - Most important options are set by default
4. Submit the restore using action button
5. load restore
   - Tip Load restore -g
   - Allows to keep the restore application loaded
   - If you have remote console permissions: → Action button to start restore
File Types involved

- ***.nsf, *.ntf, *.box**
  - Original databases to be backed up

- ***.txn**
  - Transaction log files to be backed up

- ***.DELTA**
  - Delta files created during backup – if database changes during backup (e.g. names.nsf.DELTA)

- ***.DAD**
  - Restored database before “activation” (e.g names.nsf.DAD)
  - Restored with a temporary name to avoid server will find the database before it is recovered
Restore operation in detail

- Servertask copies back database to restore location
- File name will have a .DAD extension to ensure the server does not access it meanwhile
- Disables replication and sets other settings
- Applies .DELTA file if available and recovers the database
- Renames the database to remove the .DAD extension
- Recovery documents and folders into original database if requested
Restore documents & folders to original database

- **Restore documents and folders**
  - Restore deleted documents & folders into original database!
  - Finds deletion stubs and updates it to replace the deletion stub
  - Folder notes are updated and replicated back to the original database overwriting the deletion stub
Backup Concept: Transaction Log in Archived mode

- **Full backup** + Translog Backup allows to recover data at any **point time**

- Archive Style transaction logging is enabled in Domino
  - TXN files **must** be backed up in time to recover disk space

- Typical setup:
  - Full backup (e.g. twice a week)
  - Transaction Log Backup every 2 hours
Archive Transaction Log Backup

• More complex backup setup

• Schedule a full NSF backup at least twice a week
  - Better every night to reduce the time needed for restore and for consistent backups

• Incremental backup for databases with changed DBIID (fixup -j, compact -B/-C, DBMT)

• Ensure number of compacts is reduced and always scheduled before a full backup
  - Switch to DBMT for a better control and more modern and efficient maintenance operations!
Restore “Point in time”

- Point in time restores based on the full backup of database
- Changes applied from transactions recorded in translog until point in time
- **All** translog extends between the database backup and point in time selected have to be inspected for databases to recover
- This means temporary restoring all translogs one after another from the backup if not on disk!
- Restore task takes care of the restore
  - But it can take hours depending on the number of translogs affected!
  - Ensure shorter backup cycles – One week can cause very long restores on a Friday afternoon!
Backup Operations

• Load backup -b
  – Full backup (default)

• Load backup -t
  – Translog archive backup

• Load backup -i
  – Incremental backup
  – Backup all databases with new DBIID since last full backup
  – All new databases or databases with changed DBIID (fixup -j, compact -B or -C, DBMT, ..)
Backup Integrations
Integration Points

• **File backup commands** – build-in, leveraging OS commands
  - Most easy to setup backup/restore
  - Customizable via @formulas
  - Can be also leveraged for commercial software like Cohesity with their backup repository
  - Used for backup, restore and backup prune operations (delete)

• **Command-Line interface**
  - Very flexible, standardized interface for Windows (batch) and Linux (shell script)
  - Customizable via @formulas
  - Return strings can be passed to the backup software for backup reference and status

• **Agent interface**
  - Can be for example used for REST requests leveraging HTTP Request class
Backup Scenario: Integration on Script level

- **Command Line backup with a backup application**
  - Or command line integration with a client like AWS S3 CLI or any other custom script

- **Flow**
  - Domino backup server task takes one database after another into backup mode
  - OS level command is called to backup the database
  - Bring database into normal operations mode
  - Check if changes occurred during backup and create a *delta* file with changes

- **Challenges**
  - Restore operation is required to bring a restored database online
  - Deltas need to be applied back on restore
HCL GitHub Project for Backup Integrations

- https://github.com/HCL-TECH-SOFTWARE/domino-backup
- https://opensource.hcltechsw.com/domino-backup/

Domino Backup and Restore

Domino V12 introduces a Backup & Restore which is designed for easy integration with 3rd party backup solutions.

The server tasks backup and restores are accompanied by a new DominoLite, which holds all the configuration, logs and a database inventory. A Notes UI allows you to restore databases with many options including restoring deleted documents and folders into the original database.

The solution is intended to complement existing backup solutions and to make it easier for customers and partners to integrate with existing backup solutions which are not Domino aware today. Integrated solutions from a backup vendor should always be the preferred backup option if available!

Out of the box Domino Backup is configured to use a file back-end with integrated copy operations directly performed by core Domino. This default configuration can be used with different file back-ends. But Domino Backup can also be extended using custom scripts.

Integration options and scope

Integration might consist of the following types:

- Integrated file backup operations
- Custom scripted integration
- Snapshot backups
HCL GitHub Project for Backup Integration Solutions

• Open Source repository for 3rd party backup integrations based on the framework Domino provides

• Integration point/operations documentation

• Additional information
  – Technologies like snapshot
  – Best practices around backup and storage optimization
  – File system back-end operations

• Main entry point for supporting backup integrations
  – It is not intended that HCL support should help to integrate applications

• Collaboration in the community via GitHub repository
  – Backup vendors, customers, partners and HCL
Command-Line Interface Parameters

- Example scripts for Windows & Linux

- Standardized parameter list for all backup/restore/prune operations

  - **PhysicalFileName** – physical file name
  - **FileName** – logical Notes file name
  - **BackupDbRef** – backup reference passed to the backup application
  - **BackupNode** – backup node name used for backup/restore – usually short server-name
  - **BackupName** – backup name can be specified during backup
  - **BackupMode** – FULL, INCREMENTAL, SELECTIVE, ...
  - **BackupStartDT** – Start of backup job – useful for backup identification
  - **BackupTargetDir** – Target directory (or location) for backup – specified in the configuration
  - **RetentionDays** – Backup Retention days
“Delta Files” created during backup

• The Domino Backup API brings databases into a consistent state for backup

• Databases can be backed up on file-level once backup is started

• **Important**: Changes in database are recorded and **need to be applied** to the backup database to bring the database into **consistent** state!

• Delta files can be
  - a.) merged during backup in case of file backup operations
  - b.) on restore in case of other operations

• **Delta Files have to be merged in any case!**
  - But don't worry - The Domino backup & restore takes care about tracking delta files
Implementation reference: S3 Storage Minio

- Command line integration
- Simple integration based on **formulas** not a “Cmd file”
  - Good example to show how @Formula integration works
  - Results are captured by the backup/restore task
    - OK and Error strings can be used to check operation status
    - Needs helper binary and config from S3 vendor (e.g. `mc` from Minio)
Implementation reference: S3 Storage AWS CLI S3

- Same type of integration
- Just a different command line used
  - In this case AWS CLI with S3 command
- Needs target specific configuration
  - AWS credentials and target information for the bucket used
Demo: Backup CMD Integration

Example: Backup to S3
Domino V12.0.1 Backup & Restore

• Domino V12 already had all integration points needed for building flexible integration for many types of backup back-ends

• Domino V12.0.1 Backup & Restore offers additional functionality for closer integration
  
  − Fit & Finish for existing integration options
  − Support for 3rd party restore operations
  − Better support for snapshot backup vendors like Veeam
Domino 12.0.1 3<sup>rd</sup> Party Restore

- Backup vendors with full Domino support leveraging the standard C-API interface can now leverage the Domino restore interface for restore operations.

- **Restore 3<sup>rd</sup> party restore mode**
  - Allows to integrate with your favorite Domino aware backup solution
  - Leverages same type of integration used for the existing backup integrations

- **Flow:**
  - Specify database to restore & run restore operation
  - "restore" task executes integration script to trigger 3<sup>rd</sup> party restore operation
  - 3<sup>rd</sup> party restores database and brings it online
  - Domino restore performs additional restore operations (changing replica-ID, restoring documents & folders!)
  - Existing DAOS restore operations can be combined
Configure Domino 3rd Party Restore

- Same type of integration you know from full integration operations
- Restore operation triggers the external backup vendor for a full restore including bringing the database online
- Domino restore operations will be executed immediately after restore completes
- Tip:
  - Restore time cannot be passed via Cmd operation
  - Best choice is probably “Cmd Formula”
  - Allows to pass any field in restore document
Configure Domino 3rd Party Restore

• Tip, if you only want to change the date, there is a 3rd party date formula

\[ X := \text{@If} (\text{RestoreDateTime}=""; @Now; \text{RestoreDateTime}); \\
Y := @\text{Text}(\text{@Year}(X)); M := @\text{Text}(\text{@Month}(X)); D := @\text{Text}(\text{@Day}(X)); H := @\text{Text}(\text{@Hour}(X)); N := @\text{Text}(\text{@Minute}(X)); S := @\text{Text}(\text{@Second}(X)); \\
@\text{Repeat}("0"; 4 - @\text{Length}(Y)) + Y + @\text{Repeat}("0"; 2 - @\text{Length}(M)) + M + @\text{Repeat}("0"; 2 - @\text{Length}(D)) + D + @\text{Repeat}("0"; 2 - @\text{Length}(H)) + H + @\text{Repeat}("0"; 2 - @\text{Length}(N)) + N + @\text{Repeat}("0"; 2 - @\text{Length}(S)) + S; \]
Enable 3\textsuperscript{rd} Party Restore Operations

- Open global configuration
- Enable "Restore 3\textsuperscript{rd} party"
- Brings \texttt{dominobackup.nsf} into 3\textsuperscript{rd} party operations mode
Perform 3\textsuperscript{rd} Party Restore

- 3\textsuperscript{rd} party restore can obviously not use the database inventory
- Restore operations are triggered by entering the database name manually
Tip: 3rd Party Restore → Select Databases via SmartIcon

- SmartIcons are globally available in any database

- **Idea:** Run SmartIcon on **person/mail-in** or **catalog.nsf** document to create a restore document

```plaintext
Server := @Subset(@DbName; 1);
BackupDB := "dominobackup.nsf";
@if (Server = ""; @Prompt([Ok];"Error"; "Cannot run on local database!"); "");
@if (Server = ""; @Return (""; "");
Database := @If (Mailfile != "";
@if (@Ends(@LowerCase(MailFile);".nsf");MailFile;MailFile+".nsf"); PathName !="";PathName;"");
@Command([Compose]; Server : BackupDB; "RestoreJob");
@UpdateFormulaContext;
@SetField ("ServerName"; Server);
@SetField ("PathnameRestore"; Database);
@Command([EditGotoField]; "RestorePath");
```
Domino V12.0.1 End to End Logging

• In Domino 12.0 logs for command-line/script integrations are only parsed for errors
  – For debugging you had to write the logs into files and manage them on your own

• Domino 12.0.1 backup automatically collects all standard output from invoked scripts/commands
  – When backup/restore scripts/commands fail, the output is stored in a temporary file
  – Once the backup or restore operations completes, the full logfile is appended to the backup/restore note
  – On by default and cannot be disabled
  – You can change the configuration to log all output for debugging purposes (Main configuration tab)
  – Tip: Redirect error output for scripts by adding `2>&1` direct to the command configuration
Domino V12.0.1 Selective Backup Prune

• In Domino 12.0 backups are only pruned by retention time

• Domino 12.0.1 introduces a selected prune operation directly from the Log views
  - Just select one or more backups to be pruned and run “load backup -p” (or Run Prune Server Cmd)
Domino V12.0.1 Multi Restore Improvements

• In Domino V12.0 the multi restore operation was limited
  – All databases had to come from the same backup
  – There have been some UI issues not showing the right error messages

• **Domino 12.0.1 now fully supports multi database restore**
  – Always the latest backups for a database matching the restore time, are used
  – Databases don’t need to be in the same backup (e.g. incremental backups)

• How does it work?
  – Admin selects the restore time
  – Submit logic checks each selected database for matching backup and writes the time into each doc
  – Restore operation will restore each database from the specified backup date
Domino V12.0.1 Multi Restore Optimization

• Challenge
  - Restoring multiple database from a snapshot would potentially involve many mount/unmount operations
  - For many backup vendors mounting a snapshot is a quite time consuming operation

• Solution
  - New pre-restore and post-restore scripts to allow restore operation optimization

• Example logic:
  - **Restore DB operation script** → Checks if the right snapshot is mounted
    - → If yes, just copy the database from snapshot
    - → If not mount the snapshot and copy the file
  - Next file might hit the already mounted snapshot – or mounts another snapshot
  - **Post Restore Script** → Unmounts all mounted snapshots
Veeam Backup & Replication

Reference implementation published on GitHub
Backup Scenario: Snapshot

- Works similar like previous scenario but brings all databases into backup mode at once

- Flow
  - Bring all databases into backup mode
  - Call an OS level command to take snapshot
  - Bring all databases into normal operations mode
  - Create a delta file with changes for each database with changes
  - Operations on Domino side finished, snapshot and delta files are static and any file backup can be used

- Backup using snapshot
  - Many different options depending on the solution used
  - Most simple approach on Windows – without a backup application supporting snapshots itself
    - Call a VSS snapshot command and to get a temporary snapshot
    - Use any file backup solution to backup the snapshot along with delta files
Backup Concept : Snapshot

• Flow:
  1. Backup application brings all databases into backup mode
  2. Snapshot is initiated – should not take more than a couple of sections
  3. Delta files from backup are stored
  4. Snapshot and delta files are backed up

• On restore usually the snapshot is mounted to “copy” databases back

• Delta files are usually saved to different backend or separate snapshot of another disk
Snapshot backup on Windows

• **VSS Snapshot** (Volume shadow copy) is build into Windows
  - Usually leveraged by backup applications to create a consistent state of an application
  - Domino Backup is **not a VSS writer** but brings databases into consistent state before initiating snapshot
  - Only works well if the application gives back control after the snapshot is created

• **Scenarios**
  - Snapshot enabled backup application is used
  - Snapshot aware storage/infrastructure is used
  - Domino can use native Windows tool “**diskshadow.exe**” to create snapshot and invokes backup operation

• **Different ways to use snapshot result**
  - Only use snapshot to bring databases into consistent state for backup and use classical backup
  - Use native snapshot backup in the storage backend to keep the backup as snapshot
Snapshot backup on Linux

• Most file systems used don’t have snapshot capabilities on their own

• **OpenZFS** and **btrfs** support natively support snapshots

• Some applications like Veeam use own storage drivers to ensure block level delta backup

• Storage providers like NetApp support snapshots

• Depending on the technology used different type of integrations possible
Veeam Backup and Replication 11

- Enterprise solution widely used
- Veeam named a Leader for the 4th time in a row
  - Category: “Data Center Backup and Recovery Solutions”

- Current situation
  - Simple “pre-freeze” script “drop databases” to try to bring databases into sync
  - Only crash consistent backup
  - Not supported by Domino
  - Does not support Backup API integration via agent
  - Restore requires Veeam admin to manually search and mount the right backup
  - No automation or additional functionality like disabling replication, safely bring database on-line etc.
Design Goal

- Full featured Domino Backup **end to end integration** with Veeam Backup and Replication
- **Veeam Admin:** Expects simple Domino integration with “freeze” scripts configured in Veeam
- **Domino Admin:** Expects seamless restore databases operations triggered from Domino
  - No manual mount or recovery operations in a Veeam client UI
- Secure, optimized and reliable backup solution for Domino which takes benefit of both worlds
- Support Windows and Linux back ends with
  - **VMware, Hyper-V agent less backup** on guest OS
  - Currently not in focus:
    - Servers using **Veeam agent** installed on OS level
Domino Backup

Domino on Win/Linux

Domino + Backup

Snapshot Scripts

dominobackup.nsf

VM Disks

Backup

ESXi

Veeam Backup & Replication on Windows 2019

1. Backup Job
2. Freeze Scripts SSH/Win
3. Snapshot

Linux Private key/Windows admin account

Veeam Data Store

Backup (Restore Point)
Domino Restore

Domino on Win/Linux

1. Domino + Restore
2. Restore Script
3. Private key
4. SSH
5. OpenSSH Server
6. Copy
7. VM Disks

Veeam Backup & Replication on Windows 2019

1. Configuration (JSON)
2. PowerShell
3. Veeam Backup & Replication
4. Veeam Data Store
5. ESXi
6. Mount
7. VM Disks
8. dominobackup.nsf

Backup (Restore Point)

Copy Domino on Win/Linux
Domino Configuration for Veeam Backup

- Admin deploys script on each server and imports DXL configuration
- Customized settings in “main” tab
Veeam Job Configuration

- Standard Veeam job configuration
- Includes application processing via snapshot scripts
- Requires an admin user defined in Veeam Credential store
Veeam Backup & Replication Configuration

• Veeam admin configures backup jobs with simple “pre-freeze” and “post-thaw” scripts
  – Veeam copies scripts for each backup operation to the target guest OS
  – Call defined freeze scripts on Domino server to start Domino backup to bring all databases into backup mode

• Windows:
  – Admin account for freeze-script and mount operations in Veeam credential store

• Linux:
  – SSH key in Veeam credential store used for freeze-script operations for “notes” user
  – No elevation of user required for freeze operations with notes user
  – Elevated access for root permissions using a “veeam-mount” user for restore mount operations
    ▪ Requires sudo permissions for “veeam-mount” user for all Domino Linux servers
    ▪ By design a the separate “veeam-mount” user is defined for elevated operations needed for restore mount
Credentials Configuration for Linux

- Veeam supports SSH private/public key authentication for Linux
- Ensure the key is in RSA key format for importing
- Enable account privilege elevation
- Ensure the description matches your configuration
  - Credentials are search by “Description”
- SSH public key for *veeam-mount* user on each Domino Linux server
  - Account needs *sudo* permissions
Per Server Configuration

• Backup operations are configured and scheduled using standard jobs with freeze scripts

• Restore operations require separate configuration per Domino server

• Configuration file in JSON format contains one entry per Domino server
  - Defines access and mapping to virtual machine (VM backup) or host (agent backup)
Domino Restore Operations with Veeam Integration

• **Seamless restore for Domino Administrator**

  - Domino admin uses Domino Backup Inventory to select database to restore
  - Creates a restore job
  - Domino Restore server task takes care of
    - Finding the right backup
    - Mounting the backup
    - Copying the database to the requested location
    - Unmounts the backup
    - Applies requested operations
    - Brings database on-line
# Restore Integration via PowerShell and OpenSSH Server

- Veeam Backup and Replication leverages the Windows platform

- REST Interfaces for Backup & Replication and Veeam Enterprise Manager do not provide the capabilities Domino needed for integrated restore operations

- **Veeam PowerShell interface** is the best supported and most flexible automation interface
  - Runs locally on Veeam server for all servers in backup

- Remote PowerShell is **not** an option because of security requirements and Linux platform support
  - Note: Linux meanwhile supports PowerShell but it would not be likely Linux admins would install it

- Integration is leveraging an **OpenSSH server** included and supported on Windows 2019+ / Win 10+
  - Including **Public/Private key authentication** with up to date standards (ECDSA and ed25519)
OpenSSH Server Installation on Veeam Server

- OpenSSH server is an optional component available starting with Win2019 / Win10
- Configuration example for **sshd_config** for SSH key only configuration is included in our repository
- Straightforward to install and configure
  - Ref: https://docs.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse
Security Configuration

• Integration leverages **SSH** to establish secured communication channel with **private/public key authentication**

• User with **Veeam Restore Operator** role and an **authorized_keys** configuration

• The **authorized_keys** configuration restricts the permission to a single PowerShell script building the bridge between Domino and the Veeam server

• **SSH key per server** (or for multiple servers) to secure access to the Veeam server for restore mount requests

• **PowerShell script + JSON configuration** further restricts access to restore mount operations
  - Each server is added the configuration and checked by IP address
  - Controls access to restores per server
Demo: Veeam Backup

In case we still have time..
Slides have most of the details
Build your own Lab Environment

• Free Backup for up to 10 Workloads!
  − Provides even free agents for notebooks
  − Can be used in production!
  − Some other limitations like only one backup repository

• Full featured functionality for virtual environments
  − VMware vSphere
  − Microsoft Hyper-V
  − https://www.veeam.com/backup-replication-system-requirements.html

Supported Virtualization Platforms

• Only VMware vSphere is supported!!
  − No support for the free ESXi platform!
  − Integration requires VMware storage APIs not included in the free offerings
  − You could start with a time limited trial
  − Or you are lucky and can use a corporate vSphere environment..

• There are some license changes in the Hyper-V platform
  − But for Windows evaluation copies and MSDN licenses Hyper-V is still included
Building your own Lab Environment

- My starting point was VMware vSphere evaluation version setup with a new ESXi server
  - You cannot extend the trial, so I built a

- All in one VM for testing lab environment:
  - Intel NUC running ESXi 7.x *
  - Windows 2022 VM with embedded virtualization
  - Veeam Backup & Replication 11.0.1
  - Hyper-V with two guest VMs
    - Windows 2022 with Domino 12.0.1
    - SUSE Leap 15.3 with Domino 12.0.1

*) Could be any virtualization platform with embedded virtualization support.
Additional Information
Backup Retention

- Domino Backup contains information about backup repository and ensures backup retention.
- Retention interval is written into backup logs and repository data to be used for backup retention.
- There is a separate prune operation load backup –p <days> to prune backups earlier.
- Backup retention time is also passed to back-ends during backup to support retention for other backup solutions.
- Backup retention integration to prune backups in backup back-ends:
  - Scripts to purge single databases and whole backups depending on back-end requirement.
Backup & Restore Logging

• Verbose Logging
  – load backup -v

• Debug Logging
  – Load backup –V

• Logs are written to console log
Domino Database Best Practices for Backup

• Leverage DAOS!
  – Will reduce NSF files up to 70%
  – DAOS is a simple file-backup of static files!
  – Recommended threshold: ~256 KB

• Use database design and document compression
  – Will save a lot of disk space – round 50% of the document data

• Enable NIFNSF to store index outside the NSF
  – Around 10% of the original database

• Use DBMT to maintain your databases
  – Configure 10 days and run it once per week before backup
$8.4 BILLION ENTERPRISE | 132,000 IDEAPRENEURS | 44 COUNTRIES

WATCH THE FILM
QUESTIONS?

Use the GoToWebinar Questions Pane

Please keep all questions related to the topics that our speakers are discussing!!

Unrelated Question => post at:
http://openntf.slack.com/