



relax-and-recover.org

Relax-and-Recover (ReaR) Basics

with live demo on real hardware

By

Gratien D'haese

Johannes Meixner





Ask Yourself: Mean Time to Restore Service

- After deploying a bad software update or configuration?
- After upgrading the Operating System to a faulty version?
 - On 50 servers? On 500 servers?
- After deleting the hard disk / SAN LUN of your main database?
 - After deleting 20 LUNs?
- After deleting the hard disk / SAN LUN of a Hypervisor?
 - All the LUNs of a virtualization cluster?
- After flooding the data center?



Relax-and-Recover (ReaR) as (DR) solution

- ReaR is a tool that implements a **DR work-flow** for Linux
- Basically meaning:
 - Modular framework written in Bash
 - Easy to extend to own needs
 - Easy to deploy (set up and forget)
 - Integration for various Linux technologies
 - Integration with various back-up solutions
 - Attempts to make system recovery *as easy as possible*
- ReaR runs on-line (no downtime to create a DR image)

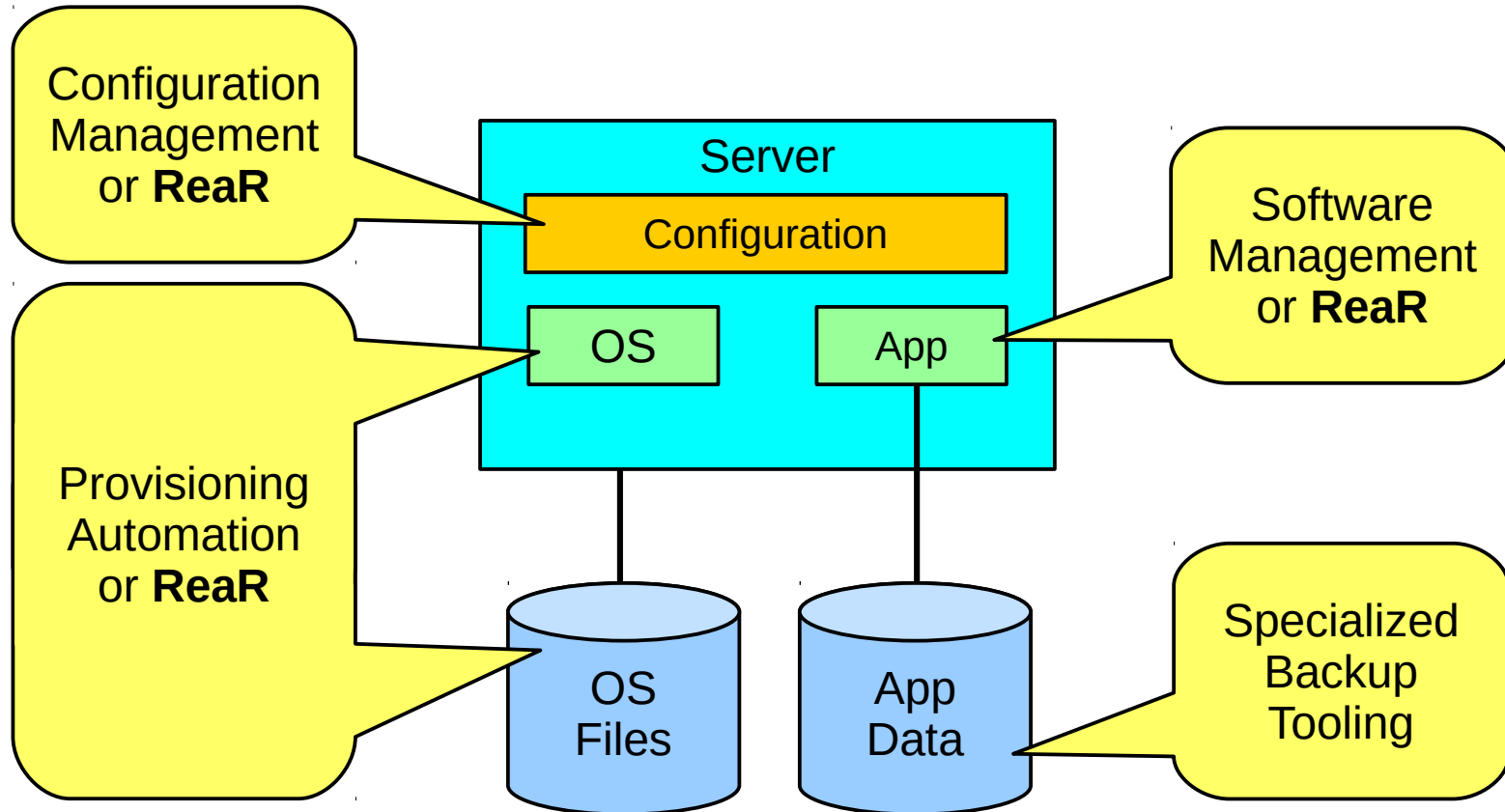


Introduction to Relax and Recover (ReaR)

- Proven solution at large enterprise customers
- ReaR established as standard solution for Linux disaster recovery in data centers
- Shipping with Fedora, SUSE and RHEL
- Integrates with many “commercial” backup software solutions, e.g. TSM, DP, NBU, NSR, ...
- Integrates with OS backup software solutions as well, e.g. GNU tar, rsync, bacula, bareos, ...
- Scales well with large amounts of servers

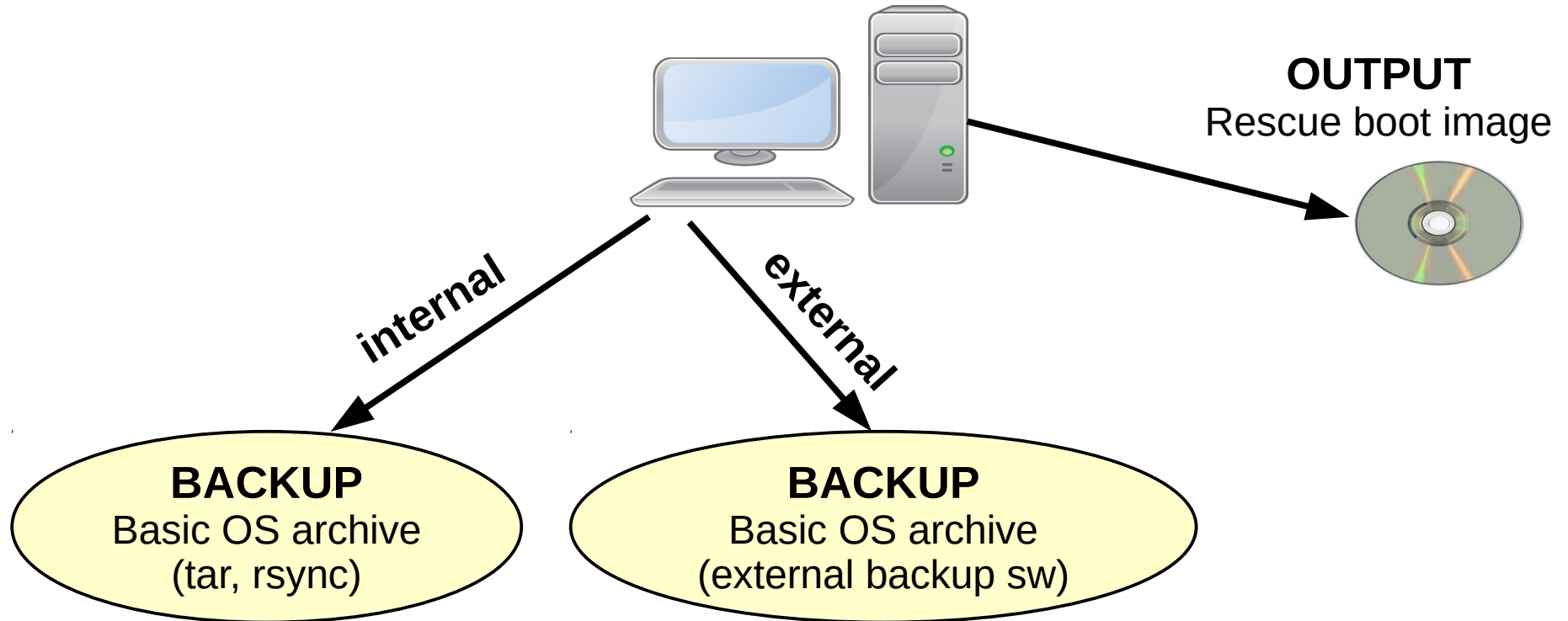


Use the Best Tool for the Restore Job





DR Flow – BACKUP and OUTPUT





Decide on DR strategy

- Which backup mechanism to use?
 - Internal backup: GNU tar, rsync
 - External backup: bacula, bareos, commercial backup solution
- Where will the backups reside?
 - NFS share, CIFS share, external USB disk, tape, local spare disk, cloud storage, DVD
 - Remote network and/or storage location
- How shall we boot the rescue image?
 - Via DVD (ISO image), tape (OBDR), network (PXE), USB disk



Basics of ReaR

- Main script is `/usr/sbin/rear`
- Shell scripts are stored under `/usr/share/rear`
- Scripts are kept together according work-flows
 - `mkrescue` (only make rescue image)
 - `mkbackup` (including make rescue image)
 - `recover` (the actual recovery part)
- Configuration via bash variables (`/etc/rear/local.conf`)
- User guide `/usr/share/doc/rear-*/relax-and-recover-user-guide.html` and man page (`"man rear"`)
- Web-site: *relax-and-recover.org*

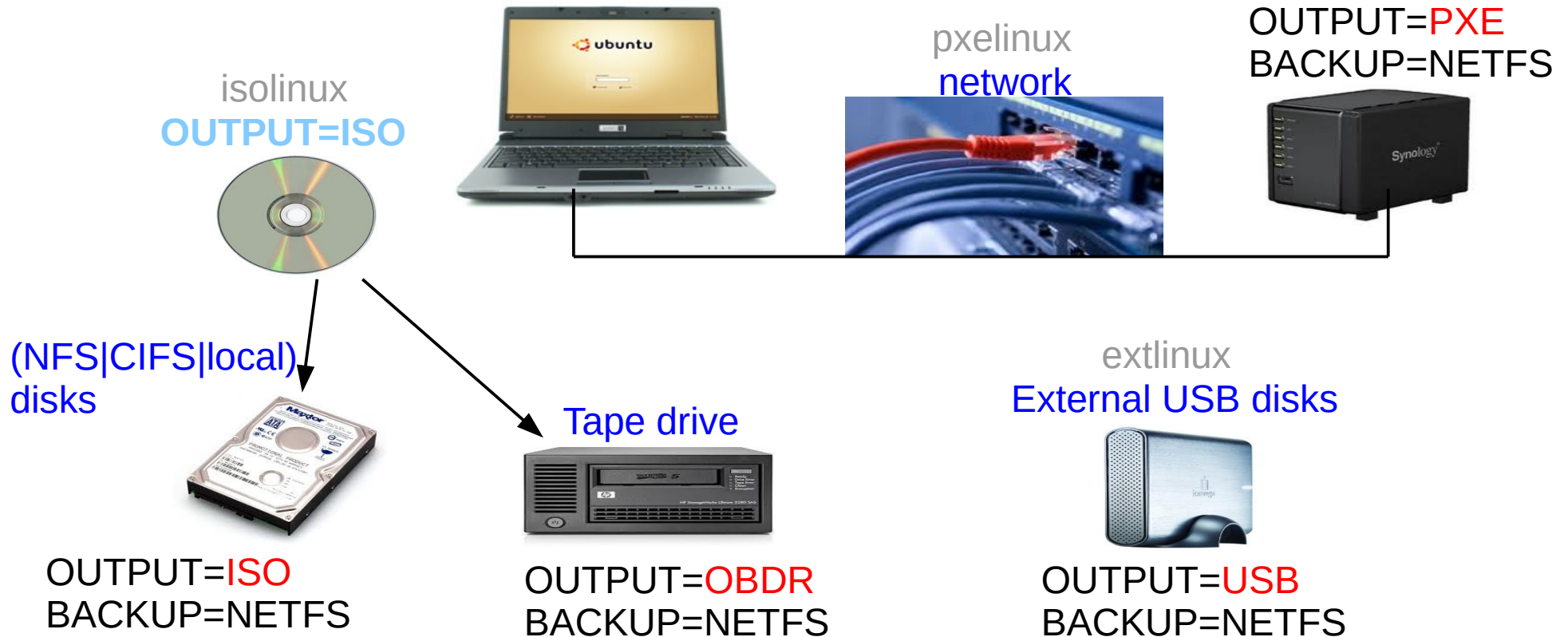


BACKUP and OUTPUT methods

- BACKUP variable defines the “**backup**” method
 - NETFS, RSYNC, DUPLICITY,
- BACKUP_URL variable defines the **location** where to store the backup archive
- OUTPUT variable defines the “**output**” method
 - ISO, PXE, OBDR, USB
- OUTPUT_URL variable defines the **location** where to store the output image (ISO image, pxe configuration, extlinux configuration)



BACKUP type NETFS





Location BACKUP_URL

- **BACKUP=NETFS**
- **BACKUP_URL** can be
 - File type: `BACKUP_URL=file:///directory/`
 - NFS type: `BACKUP_URL=nfs://nfs-server/directory/`
 - CIFS type: `BACKUP_URL=cifs://samba/directory/`
 - USB type: `BACKUP_URL=usb:///dev/disk/by-label/REAR-000`
 - ISO type: `BACKUP_URL=iso://backup`
 - Tape type: `BACKUP_URL=tape:///dev/nst0`



Backup Program

- BACKUP=NETFS
- `/usr/share/rear/conf/default.conf`
 - Default: `BACKUP_PROG=tar`
 - However, `BACKUP_PROG=rsync` is possible for local attached storage
 - `BACKUP_PROG_COMPRESS_OPTIONS="--gzip"`
 - `BACKUP_PROG_COMPRESS_SUFFIX=".gz"`
 - `BACKUP_PROG_EXCLUDE=('/tmp/*' '/dev/shm/*')`



Example Configuration Files

ls /usr/share/rear/conf/examples/

borg-example.conf

RHEL6-NETFS-of-NBU-master-example.conf

SLE11-ext3-example.conf

SLE12-SP1-btrfs-example.conf

PXE-booting-example-with-URL-style.conf

RHEL7-ISO-Oracle-example.conf

SLE11-SLE12-SAP-HANA-UEFI-example.conf

SLE12-SP2-btrfs-example.conf

rescue-and-backup-on-same-ISO-image-example.conf

RHEL7-PPC64LE-Multithread-PXE-GRUB.conf

SLE12-btrfs-example.conf

USB-and-Samba-example.conf



ReaR 'dump' Workflow

Configuration tree:

```
Linux-i386.conf : OK
  GNU/Linux.conf : OK
    Fedora.conf : missing/empty
      Fedora/i386.conf : missing/empty
        Fedora/7.conf : missing/empty
          Fedora/7/i386.conf : missing/empty
            RedHatEnterpriseServer.conf : missing/empty
              RedHatEnterpriseServer/i386.conf : missing/empty
                RedHatEnterpriseServer/7.conf : missing/empty
                  RedHatEnterpriseServer/7/i386.conf : missing/empty
                    site.conf : OK
                      local.conf : OK
```





Configuration /etc/rear/site.conf Example

```
COPY_AS_IS=( "${COPY_AS_IS[@]}" /etc/install \  
/usr/bin/perl /usr/lib64/perl5/CORE/libperl.so \  
/usr/bin/seq /sbin/lspci )  
RESULT_FILES=( $VAR_DIR/sysreqs/Minimal_System_Requirements.txt )  
  
OUTPUT=ISO
```



Configuration /etc/rear/local.conf Example

```
ONLY_INCLUDE_VG=( "vg00" )  
BACKUP=NETFS  
NETFS_URL=nfs://nas.example.com/vol/linux/linux_images_1/node05.example  
.com  
NETFS_PREFIX=image  
NETFS_KEEP_OLD_BACKUP_COPY=y
```




/etc/rear/local.conf Docker Node Example

```
# ONLY_INCLUDE_VG=( "vg00" )
BACKUP=NETFS
BACKUP_URL=nfs://nas.example.com/vol/linux/linux_images_1/node69.example.com
BACKUP_PROG_EXCLUDE=( "${BACKUP_PROG_EXCLUDE[@]} '/app/docker/*'
'/app/example/docker/*' )
NETFS_PREFIX=image
NETFS_KEEP_OLD_BACKUP_COPY=yes
AUTOEXCLUDE_DISKS=no
CLONE_USERS=( "${CLONE_USERS[@]}" oracle )
CLONE_GROUPS=( "${CLONE_GROUPS[@]}" dba )
SSH_ROOT_PASSWORD="relax"
TIMESYNC=NTP
TIMESYNC_SOURCE=10.10.10.10
COPY_AS_IS=( "${COPY_AS_IS[@]}" /etc/oratab clear )
```



/etc/rear/local.conf (SAP DB System Example)

```
# ONLY_INCLUDE_VG=( "vg00" )
BACKUP=NETFS
BACKUP_URL=nfs://nas.example.com/vol/linux/linux_images_1/node41.example
.com
BACKUP_PROG_EXCLUDE=( ${BACKUP_PROG_EXCLUDE[@]} '/DBEXPORT/*'
'/oracle/*/mirr*' '/oracle/*/or*' '/oracle/*/sap*' '/oracle/*/flash*' )
NETFS_PREFIX=image
NETFS_KEEP_OLD_BACKUP_COPY=yes
AUTOEXCLUDE_DISKS=no
CLONE_USERS=( "${CLONE_USERS[@]}" oracle )
CLONE_GROUPS=( "${CLONE_GROUPS[@]}" dba )
SSH_ROOT_PASSWORD="relax"
TIMESYNC=NTP
TIMESYNC_SOURCE=10.10.10.10
COPY_AS_IS=( "${COPY_AS_IS[@]}" /etc/oratab clear )
```



/etc/rear/local.conf (DB System Example)

```
# ONLY_INCLUDE_VG=( "vg00" )
BACKUP=NETFS
BACKUP_URL=nfs://nas.example.com/vol/linux/linux_images_1/node21.example.com
BACKUP_PROG_EXCLUDE=( "${BACKUP_PROG_EXCLUDE[@]} '/u02/ora*'
'/u02/recoveryarea01' )
NETFS_PREFIX=image
NETFS_KEEP_OLD_BACKUP_COPY=yes
AUTOEXCLUDE_DISKS=no
CLONE_USERS=( "${CLONE_USERS[@]}" oracle )
CLONE_GROUPS=( "${CLONE_GROUPS[@]}" dba )
TIMESYNC=NTP
TIMESYNC_SOURCE=10.10.10.10
SSH_ROOT_PASSWORD="relax"
COPY_AS_IS=( "${COPY_AS_IS[@]}" /etc/oratab clear )
POST_RECOVERY_SCRIPT=/mnt/local/u02/restore_oracle_u02_database_directory.sh
```



Live ReaR recovery on real hardware



NOT ALL BACKUP IS CREATED EQUAL