

Essentials

about

Disaster Recovery (abbreviated DR)

with

Relax-and-Recover (abbreviated ReaR)

Johannes Meixner

<jsmeix@suse.com>



Why DR with ReaR matters

Relax-and-Recover is currently used in particular by business/enterprise users for disaster recovery on thousands and thousands of various systems.

Topics

What does disaster recovery mean ?

How does disaster recovery work ?

What disaster recovery is not meant to do

What is Relax-and-Recover ?

How does Relax-and-Recover work ?

How to set up Relax-and-Recover

Relax-and-Recover advantages/disadvantages

Relax-and-Recover limitations

Bottom line

What does disaster recovery mean ?

system got destroyed

messed up essential files (`rm -rf /lib*`), file systems, disks,...

broken hardware (partially or completely)

disaster recovery means reinstalling from scratch

on same hardware

on fully compatible replacement hardware

more than restoring files (more than backup)

1. prepare hard disk (storage)
2. restore files (from backup)
3. install boot loader

How does disaster recovery work ?

while system is up and running

save the system's payload: create backup of the files

prepare reinstalling from scratch: create recovery system

recovery system = recovery installation system + recovery installer

after system was destroyed

boot recovery installation system (on replacement hardware)

run recovery installer to reinstall system from scratch

1. prepare storage (disk partitioning, file systems, mount points)
2. store the system's payload (i.e. install files: e.g. restore a backup)
3. install boot loader

in this text “installation” and “installing” mean those three steps and “installer” means a program that essentially performs those three steps



What DR is not meant to do

no system configuration

recovery installer reinstalls system as it was before

1. disk partitions, file systems, mount points as they were before
2. configuration files as they were before (restored from backup)
3. boot loader as it was before

except very limited adaptations by recovery installer

some hardware specific adaptations (e.g. UUID in `/etc/fstab`)

no system migration

no different hardware (or different kind of virtual machine)

no different architecture (or bitness or BIOS versus UEFI)

no competition with configuration or migration tools

What is Relax-and-Recover ?

disaster recovery framework

no disaster recovery solution that "just works"

complements backup

backup via external software that is only called by ReaR

ReaR supports "tar" and various third party backup software

for experienced users (system administrators)

command line tool "rear" + config file (no GUI)

pure bash scripts (nothing binary)

bash is the native language for system administration

meant to be adapted and enhanced as needed



How does Relax-and-Recover work ?

while system is up and running

set up ReaR: edit `/etc/rear/local.conf` (cf. next slide)

run ReaR recovery system builder: **rear -d -D mkbackup**

makes recovery installation system + recovery installer (ISO image)

the resulting **recovery system is specific for one particular system**

calls external tool to backup files (by default a “tar.gz” on NFS)

after system was destroyed

boot recovery installation system (on compatible hardware)

run recovery installer: **rear -d -D recover**

1. prepares storage: partitioning, file systems, mount points
2. stores the payload: calls external tool to restore files (by default “tar”)
3. installs boot loader

How to set up Relax-and-Recover

how to do the recovery process

`/etc/rear/local.conf` (e.g. how to make backup and ISO image)

examples: `/usr/share/rear/conf/examples/*.conf`

defaults: `/usr/share/rear/conf/default.conf`

what to recreate (e.g. partitioning, file systems,...)

partially via `/etc/rear/local.conf` (e.g. what to exclude)

partially by editing the scripts (e.g. new stuff to include)

what the recovery process actually does

editing the scripts

adapt how the recovery system gets generated

adapt what the recovery installer actually does



Relax-and-Recover advantages

but each becomes a disadvantage from another point of view

generic (all of ReaR is pure bash scripts)

can be relatively easily adapted and enhanced by the user

but often the scripts must be adapted and enhanced

ReaR updates (e.g. package updates) may overwrite adapted scripts

small (system specific generated recovery system)

but hopeless when ReaR installation system fails to boot

fast (system specific working recovery installer)

ReaR installer directly runs low-level tools (parted, mkfs,...)

but when it fails it is a dead end for unexperienced users

one-time band-aid fixes relatively easily possible for experienced users

ReaR installer scripts can be adapted from within the ReaR installation system



Relax-and-Recover limitations

There is no such thing as a disaster recovery solution that "just works".

limited to what the ReaR recovery system can do

- ReaR installation system and SUSE inst-sys totally different

 - ReaR installation system may fail to boot where SUSE inst-sys had worked

- ReaR installer and SUSE installer (Auto)YaST totally different

 - ReaR installer may fail where SUSE installer had worked

 - ReaR installer may reinstall with (possibly subtle but severe) differences

cope with Relax-and-Recover limitations

- verify installation by ReaR works and results what is intended

- do actual productive deployment by ReaR (proves the former)

- be prepared for manual intervention (know your system)

 - know how to install directly with low-level tools (parted, mkfs,...)



Bottom line

What matters in the end

Regardless how a system was installed and
regardless what is used for disaster recovery
eventually

a disaster recovery installation
will be the final system installation.

The final installer is the disaster recovery installer.

As a general public accessible entry point
visit the openSUSE Wiki page

SDB:Disaster Recovery

http://en.opensuse.org/SDB:Disaster_Recovery

Thank you.





Corporate Headquarters
Maxfeldstrasse 5
90409 Nuremberg
Germany

+49 911 740 53 0 (Worldwide)
www.suse.com

Join us on:
www.opensuse.org

Unpublished Work of SUSE LLC. All Rights Reserved.

This work is an unpublished work and contains confidential, proprietary and trade secret information of SUSE LLC. Access to this work is restricted to SUSE employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of SUSE. Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

General Disclaimer

This document is not to be construed as a promise by any participating company to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. SUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for SUSE products remains at the sole discretion of SUSE. Further, SUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All SUSE marks referenced in this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.

