

Network Working Group
Internet-Draft
Intended status: Informational
Expires: May 25, 2010

D. Brasher
Interlinux LTD
November 21, 2009

Distributed Internet Archive Protocol (DIAP)
draft-brasher-diap-09

Abstract

A de-centralised, self-contained and managed storage protocol. A system to provide strong storage fail over by using existing resources over networks distributing vital data evenly. Rapid deployment and high redundancy for small to medium organisations as well as individuals. Designed to reduce dependency on tape backup systems. The protocol also has implications for long term archiving. By classifying data vitality values the limitations in physical space due to bandwidth constrictions can be overcome and the usefulness of DIAP maximised.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on May 25, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Table of Contents

1. Introduction	3
2. Architecture	3
3. Extended Data Retention	8
4. Fill mechanism	8
5. Prototype Design	9
6. Hyper Virtual auto-changer	9
7. Data Vitality	9
8. DIAP Rule of Thumb	9
9. Community Project and UK Trademark	10
10. DPA	10
11. Conclusion	10
12. Security Considerations	11
13. Acknowledgements	11
14. Change Log	11
15. Informative References	11
Author's Address	12

1. Introduction

Three nodes either between sites say between offices, homes, on a campus or over WAN's, which could be dedicated to storage or used for existing services, have a round robin synchronisation of FULL - differential backup pools where the source of data ranges from a personal laptop to a file store over unused band-width where the data rate is dynamically controlled, including compression, according to load and availability. Three for simplicity and because the probability of failure beyond three is so small the extra coding to accommodate more nodes would be self-defeating. In real life use of three nodes for a DIAP pool is strong enough. Chaining together DIAP pools to extend data retention periods is a future aim of the project. Also designed, as project maturity is reached, to help reduce an organisations carbon footprint, the extent to which is unknown at this stage.

2. Architecture

The system reduces single point of failure by creating a single FULL copy on each node at the beginning of the month the storing the differentials in a distributed manner. Use a program such as Bacula set to use a monthly FULL - differential. If a copy fails then the system will retry the next day but you loose the day of failure. Using rsync logs you can trace / track the successful copies. The copies are staggered so that each rsync copy list is made before new files are put into each directory by a few minutes. I.e. bd9-cd9 starts before ad9-bd9 and ad0-bd0 is last. Redundancy is split across three nodes, no duplicate days apart from the first FULL copy. DIAP pool is possibly equivalent to 30 tapes every month but stored at three locations. 10 days every three days, at each. You are advised to have some knowledge of the average differential size.

Slots

Slots	A	B	C
(Dirs)	aFull01	bFull01	cFull01
	aFull02	bFull02	cFull02
	ad00		
	ad01	bd01	cd01
	ad02	bd02	cd02
	ad03	bd03	cd03
	ad04	bd04	cd04
	ad05	bd05	cd05
	ad06	bd06	cd06
	ad07	bd07	cd07
	ad08	bd08	cd08
	ad09	bd09	cd09
	ad10	bd10	cd10
	ad11	bd11	cd11
	ad12	bd12	cd12
	ad13	bd13	cd13
	ad14	bd14	cd14
	ad15	bd15	cd15
	ad16	bd16	cd16
	ad17	bd17	cd17
	ad18	bd18	cd18
	ad19	bd19	cd19
	ad20	bd20	cd20
	ad21	bd21	cd21
	ad22	bd22	cd22
	ad23	bd23	cd23
	ad24	bd24	cd24
	ad25	bd25	cd25
	ad26	bd26	cd26
	ad27	bd27	cd27
	ad28	bd28	cd28
	ad29	bd29	cd29

Table 1: Slots

Calculations:-

LBM = Lowest Maximum Bandwidth between any three nodes NB: actual max transfer will vary so test transfers are recommended for accuracy.

LBM assumes all available bandwidth is allocated to running DIAP.
 Max aFULL01 = LMB x 6 hrs This assumes no transfer interruptions and

that the maximum bandwidth is constant.

$$\text{Ave. Diff} = (\text{Sum } 29 \text{ (or a month) Daily Differentials}) / 29.$$

Ave Differential is variable depending on your storage growth, this represents a trend and can be an estimate to start with, but by watching the trend of Differential growth more accurate calculations can be made. It is assumed your Differentials are always less the the initial FULL copy.

$$\text{Min DIAP Dir size (node a)} = (\text{Max aFULL01} \times 2) + (29 \times \text{Ave. Diff}) + (1 \times \text{Ave. Diff}) \text{ Plus } 1 \times \text{Ave. Diff} \text{ to account for ad0.}$$

$$\text{Min DIAP Dir size (node b/c)} = (\text{Max aFULL01} \times 2) + (29 \times \text{Ave. Diff})$$

You can include transfer log files in the Min DIAP Dir size, for simplicity they have been omitted.

Example System

Example System	LMB x 6 hrs	Ave. Diff	Max aFULL01
LBM occurs between b->c	1Mbit/Sec	Est. 500 MiB	2.6 GiB

Table 2: Example System

$$\text{Min DIAP Dir size (node b-c)} = (2.6 \times 2) + (29 \times 0.5) = 19.7 \text{ GiB}$$

DIAP Dir: this is the working directory used on each node and contains all DIAP configuration working and storage directories.

If a copy fails then the system will retry the next day but you loose the day of failure. Using rsync logs you can trace / track the successful copies

Flow of data.

Day - Time	A	B
D1-T=0	aFull01->cFull01	
D2-T=0		
D2-T=0	aa00->ad01	
D2-T=0	aa00->bd01	
D2-T=3		bd01->cd01
D3-T=0	aa00->ad02	
D3-T=0	aa00->bd02	

D3-T=3		bd02->cd02
D4-T=0	aa00->ad03	
D4-T=0	aa00->bd03	
D4-T=3		bd03->cd03
D5-T=0	aa00->ad04	
D5-T=0	aa00->bd04	
D5-T=3		bd04->cd04
D6-T=0	aa00->ad05	
D6-T=0	aa00->bd05	
D6-T=3		bd05->cd05
D7-T=0	aa00->ad06	
D7-T=0	aa00->bd06	
D7-T=3		bd06->cd06
D8-T=0	aa00->ad07	
D8-T=0	aa00->bd07	
D8-T=3		bd07->cd07
D9-T=0	aa00->ad08	
D9-T=0	aa00->bd08	
D9-T=3		bd08->cd08
D10-T=0	aa00->ad09	
D10-T=0	aa00->bd09	
D10-T=3		bd09->cd09
D11-T=0	aa00->ad10	
D11-T=0	aa00->bd10	
D11-T=3		bd10-cd10
D12-T=0	aa00->ad11	
D12-T=0	aa00->bd11	
D12-T=3		bd11->cd11
D13-T=0	aa00->ad12	
D13-T=0	aa00->bd12	
D13-T=3		bd12->cd12
D14-T=0	aa00->ad13	
D14-T=0	aa00->bd13	
D14-T=3		bd13->cd13
D15-T=0	aa00->ad14	
D15-T=0	aa00->bd14	
D15-T=3		bd14->cd14
D16-T=0	aa00->ad15	
D16-T=0	aa00->bd15	
D16-T=3		bd15->cd15
D17-T=0	aa00->ad16	
D17-T=0	aa00->bd16	
D17-T=3		bd16->cd16
D18-T=0	aa00->ad17	
D18-T=0	aa00->bd17	
D18-T=3		bd17->bd17
D19-T=0	aa00->ad18	
D19-T=0	aa00->bd18	

D19-T=3		bd18->bd18	
D20-T=0	aa00->ad19		
D20-T=0	aa00->bd19		
D20-T=3		bd19->bd19	
D21-T=0	aa00->ad20		
D21-T=0	aa00->bd20		
D21-T=3		bd20->cd20	
D22-T=0	aa00->ad21		
D22-T=0	aa00->bd21		
D22-T=3		bd21->cd21	
D23-T=0	aa00->ad22		
D23-T=0	aa00->bd22		
D23-T=3		bd22->cd22	
D24-T=0	aa00->ad23		
D24-T=0	aa00->bd23		
D24-T=3		bd23->cd23	
D25-T=0	aa00->ad24		
D25-T=0	aa00->bd24		
D25-T=3		bd24->cd24	
D26-T=0	aa00->ad25		
D26-T=0	aa00->bd25		
D26-T=3		bd25->cd25	
D27-T=0	aa00->ad26		
D27-T=0	aa00->bd26		
D27-T=3		bd26->cd26	
D28-T=0	aa00->ad27		
D28-T=0	aa00->bd27		
D28-T=3		bd27->bd27	
D29-T=0	aa00->ad28		
D29-T=0	aa00->bd28		
D29-T=3		bd28->cd28	
D30-T=0	aFull01->aFull02		
D30-T=0		bFull->bFull02	Node C
D30-T=0			cFull->cFull02
D30-T=0	ad00->ad29		
D30-T=0	ad00->bd29		
D30-T=3		bd29->cd29	

Table 3: Data Flow

Start 00:00 - End 00:06 - T = 0(00:00) - T=3(03:00)

All copies from a-a are not used in bandwidth

Two entry points, aFull01 beginning of month and ad00 for the remaining days. Assuming entry points are filled during the day before the cycle begins at night.cron jobs split between 3 nodes,

ad00is cleared after copy to bd\$ The system reduces single point of failure by creating a single FULLcopy on each node at the beginning of the month then at the end of the month to cover the next 30day diap cycle. Storing the differentials in a distributed manner. Use a program such as Bacula set to use a monthly cycle. The copies between a-a and a-b occur in the first three hours then the copy from b-c happens after three hours. These times can be changed as required. Because the last nightly copy occurs between b and c, use of node c is optional.

Nightly copies to and between nodes are made to new directories, if due to some failure a node is unavailable then the copy is not made, however the next day when communication is restored copies continue to the next nightly directory. This increases robustness over previous layout as the next nightly copy is not dependent on the success of the previous night's copy. Only two copies between nodes are made between days 3-30, day 1 a single FULL and day 2 FULL and Day 30 does make an internal copy on all nodes.

3. Extended Data Retention

Each of the slots, directory structures and data flow mechanism will be reside in sub directories:

nodeX -> n*year -> 12*month -> (DIAP slot and data flow mechanism)

Retention period is defined by copy strategy, i.e. add additional monthly cycle(s) for an extra month(s) of retention up to 12 for a year. If more than one year is required then a new n*year -> 12*month -> (DIAP slot and data flow mechanism).

4. Fill mechanism

The filling mechanism works as follows: The start time is an integer between 0 and 11. The fill is triggered by a scheduling application like cron. Then a check is made to see if the previous days copy was successfully. If not then an alert is made and logged for later use. If yes then a search, using a pre-defined string, is made in a directory containing the backup volumes. If FULL volumes have been selected for collection then a check for the day of month is made. [Currently in implementation this is day 2 - this will be settable to any day]. The name of the FULL volume is a pre-defined string. If a FULL needs to be transferred then the most recent FULL volume is located. A check to see if the FULL has been collected before is made. If no then the FULL is copied to the appropriate slot and a date and shalsum created and located in the slot with the volume.

The activity is logged and the algorithm ends. If yes then the activity is logged and the algorithm stops. If a FULL volume is not required to be transferred then the most recent differential volume is located using a pre-defined string. The contents of d0 are cleared. A check is made to verify the differential has not been collected before. If yes then the activity is logged. if not the the latest differential is copied to the appropriate slot and a date and shalsum created and located in the slot with the volume. Activity is logged and the algorithm ends.

5. Prototype Design

The prototype is built of several components and uses the Linux Operating system. Bash scripts are used to deploy DIAP on three POSIX user accounts using expect and ssh. Ssh certificates are setup between three POSIX accounts. A single configuration file is use to set environment variables.

The system requires a series of directories used to store the data fed into ad0 and aFull:

```
mkdir aFull ad0 ad1 ad2 ad3 ad4 ad5 ad6 ad7 ad8 && touch log_a
```

Cron jobs are used to implement table 2 architecture:

```
0 1 0 * * rsync -az -e ssh --timeout=1800 --numeric-ids \ --log-  
file=/home/diap/log_b --ignore-errors --bwlimit=128 \ ~/aFull/ diap@  
$IP_ADD_B:bFull
```

6. Hyper Virtual auto-changer

This term is derived from the term virtual auto-changer. A virtual auto-changer still requires hardware tape drives, 'Hyper' takes this one stage further by emulating the virtual auto-changer in software.

7. Data Vitality

Data vitality is a measure of the organisation subjective view of the value of particular data types.

8. DIAP Rule of Thumb

Observing an email archive, at 272MBytes, having never deleted an email permanently and the file, ../mail, has been in use for 4 years.

During this time available xDSL line Bandwidth has increased, 2004 500MBits/sec to 1GBit/sec, 2008 1GBit/sec to 6GBits/Sec this is about a 150% yearly increase whereas the mailbox has increased yearly by about 50%. It is this difference which DIAP attempts to use classing email record as 'mission critical' - Other record types will increase at different rates, as will bandwidth depending on location, but probably less than the average yearly bandwidth increase. This idea needs expanding but forms the foundation for the usefulness of DIAP, describing a DIAP rule of thumb. DIAP can also be viewed as a technique.

9. Community Project and UK Trademark

A community project resides at <http://www.diap.org.uk> to facilitate the development of working implementations. The current working prototype is released here under GPL licence rules. A UK Trade Mark has been applied for to protect the acronym DIAP for use by the wider Open Source community.

10. DPA

DPA compliance and awareness.

11. Conclusion

The incremental data retention tuned to the needs of an organisation so that some data is always available from any node in the backup pool quickly to within a certain time frame and perhaps tape storage stations strategically places in various secure locations for older data retention. This system would avoid using prohibitively expensive packages by reusing resources, building on Open Source technologies and have a coherent strategy across many sites increasing the level of redundancy to a high degree. A three tier strategy involving DIAP as the bottom layer, file collection uppermost and use of pre-existing mid-term infrastructure could make up a disaster recovery plan.

With layers of indexing, accounting and management facilities. An assumption is that individual file encryption the responsibility of the file owner, this does not rule out hard drive or partition encryption of individual nodes considered to reside at insecure locations. If used for these locations physical security automatic fail-safe measures to trigger archive deposits useless upon theft can be deployed. Similar fail-safe techniques deployed for attempted network security breaches. Virus scanners would be set to scan

existing archives periodically and on entry to the archive pool.

12. Security Considerations

Open root access is not recommended for SSH. Using ports other than the default 22 is advised.

13. Acknowledgements

Thanks are due to Myles McClelland and a number of individuals from various groups. Also Stephen Pelc of MPE Forth for SME deployment context advice and IPR consultancy.

JISC for providing technical development funding through OMII-UK and ECS (Southampton University) in collaboration with Interlinux LTD.

14. Change Log

09 Nov 09 - change log correction.

09 Nov 09 - Fill algorithm described.

27 Jul 09 - Spell check.

27 Jul 09 - Remove section to avoid IP infringement.

15 Apr 09 - Extended data retention.

03 Dec 08 - Corrected Architecture.

02 Dec 08 - Refined Architecture.

06 July 08 - Architecture - arithmetic. Acknowledgements.

May 08 - Address.

15. Informative References

[RFC4810] Wallace, C., Pordesch, U., and R. Brandner, "Long-Term Archive Service Requirements", RFC 4810, March 2007.

[DIAP] Brasher, D., "Distributed Internet Archive Protocol (DIAP)", Nov 2009, <<http://www.diap.org.uk>>.

Author's Address

Damian Brasher
Interlinux LTD
PO Box 1623
Southampton, Hampshire S015 9AE
United Kingdom

Email: dbrasher@interlinux.co.uk

