

## Intro

EMC Data Protection Advisor (DPA) - программное обеспечение для управления защитой данных, автоматизация процессов мониторинга, анализа и составления репортов для Backup и Recovery окружений (EMC NetWorker, Avamar, IBM TSM).

Пример, на сервере NetWorker локально устанавливается DPA агент для сбора бекап данных с используемых Backup client-ов и устройств ввода-вывода, сбор данных также может быть осуществлен удаленным агентом. DPA Discovery Wizard добавит NetWorker хост с конфигурацией в DPA, информация сохранится в базе данных и будет доступна и визуализирована с помощью репортов.

## DPA 6 Стек технологий (Кластер)

Agent Framework - содержит как Си (NetWorker), так и Java модули (Sap Hana), загружаемые по запросу от определенных устройств. Агент общается с app сервером для сбора данных с аппаратных устройств. DPA использует Postgres в качестве реляционного хранилища данных.

Application сервер- stateless Java EE приложение, легко масштабируемое в кластере или при standalone инсталляции на удаленных машинах. Jboss WildFly предоставляет REST api и стек, hibernate и драйвер JDBC, который обращается к базе данных Postgres.

DPA datastore- база данных Postgres 9.6.1. Единственное, что мы изменили, это обернули его конфигурацией под наш DPA. В остальном это тот же Postgres, но все управление происходит через команды dra. Мы используем пул соединений, поэтому оставляем соединения открытыми на сервере.

В 6х версии DPA перешли к архитектуре единого монолитного процесса, проще масштабируется в кластере, но меньше видимость в отдельных сервисах при поиске ошибок .

## DCE Описание задачи

Customer: Американская фармацевтическая корпорация, дистрибьютор лекарственных средств. Обладает обширной системой администрирования сетевой инфраструктуры. Для резервного копирования и репликации критичных данных использует backup окружения, около сотни networker и tsm хостов. Используют DPA для сбора информации о бекапах в режиме реальном времени.

Issue: Datastore службы не запускаются, datastore log и application log ОС указывают, что датастор поврежден.

Root Cause: Customer - Environment

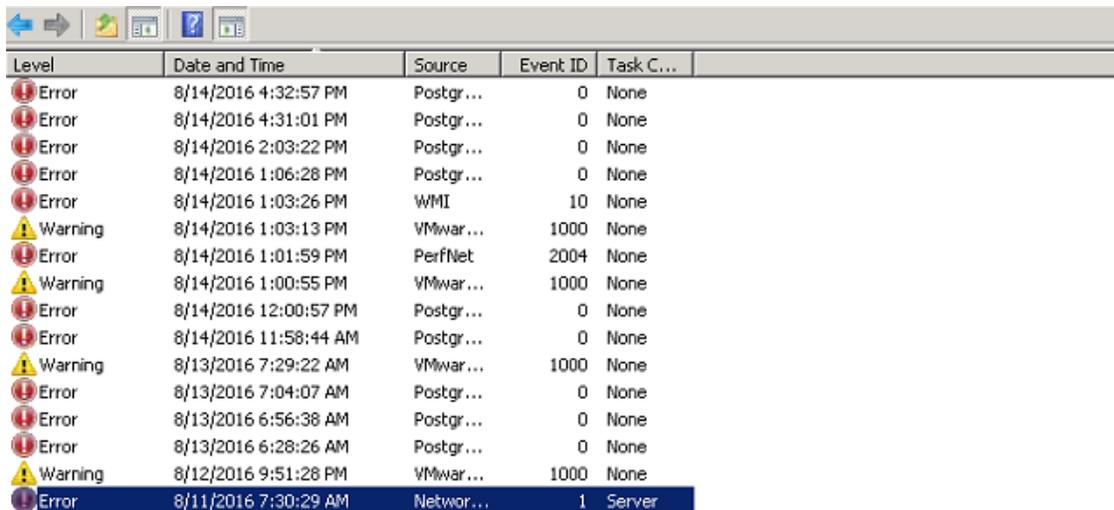
OS Platform: All

Description: Split install Window server 2008 R2

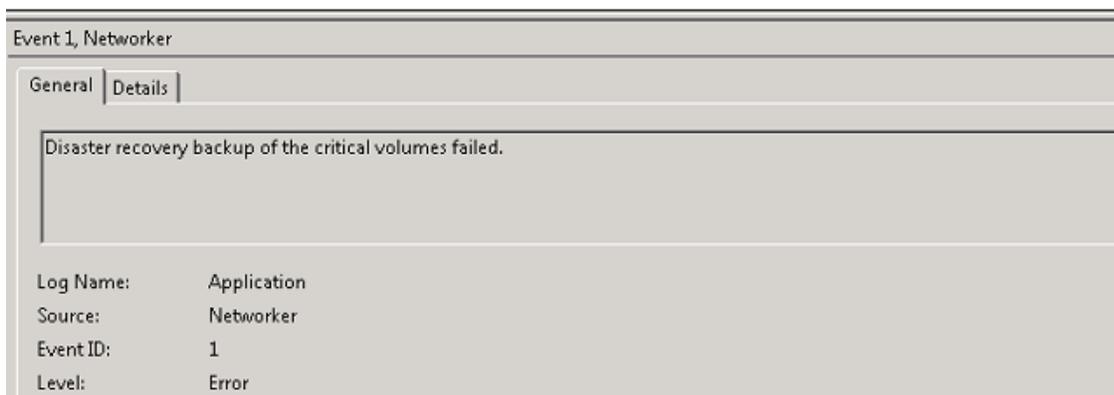
Последний успешный экспорт DS был сделан 5 июля.

После этого клиент запускал средства резервного копирования NetWorker для серверов DPA. Просмотрев application log ОС, обнаружили, что резервное копирование Networker было неудачно 11 августа, клиент подтвердил, что в это время случился сбой VSS службы Windows резервной копии хоста. Как правило, клиент не должен запускать резервное копирование средствами файлового бэкапа при работающей базе данных.

Capture Event Viewer NW backup:



Level	Date and Time	Source	Event ID	Task C...
Error	8/14/2016 4:32:57 PM	Postgr...	0	None
Error	8/14/2016 4:31:01 PM	Postgr...	0	None
Error	8/14/2016 2:03:22 PM	Postgr...	0	None
Error	8/14/2016 1:06:28 PM	Postgr...	0	None
Error	8/14/2016 1:03:26 PM	WMI	10	None
Warning	8/14/2016 1:03:13 PM	VMwar...	1000	None
Error	8/14/2016 1:01:59 PM	PerfNet	2004	None
Warning	8/14/2016 1:00:55 PM	VMwar...	1000	None
Error	8/14/2016 12:00:57 PM	Postgr...	0	None
Error	8/14/2016 11:58:44 AM	Postgr...	0	None
Warning	8/13/2016 7:29:22 AM	VMwar...	1000	None
Error	8/13/2016 7:04:07 AM	Postgr...	0	None
Error	8/13/2016 6:56:38 AM	Postgr...	0	None
Error	8/13/2016 6:28:26 AM	Postgr...	0	None
Warning	8/12/2016 9:51:28 AM	VMwar...	1000	None
Error	8/11/2016 7:30:29 AM	Networ...	1	Server



Event 1, Networker

General Details

Disaster recovery backup of the critical volumes failed.

Log Name: Application  
Source: Networker  
EventID: 1  
Level: Error

Datastore начал сообщать о повреждении с 13 августа, 6:27 утра:

```
2016-08-13 06:30:27.975 BST 6680 57aeb067.1a18 8 :FATAL: terminating connection because of crash of another server process
```

```
2016-08-13 06:30:27.975 BST 6680 57aeb067.1a18 8 :FATAL: connection to client lost
```

После выключения базы, app сервер выполнил три попытки успешно завершить обработку данных, неуспешно из-за разрыва соединения с датастором. dpa ds start debug- output

Поскольку последний успешный экспорт был сделан 5 июля, что слишком давно, клиент не может позволить себе откатиться настолько далеко, придется устранять проблемы с текущим датастором. Резервное копирование файловой системы NW на сервере DS в настоящее время отключено (14 августа). Сохранена копия текущих данных \$PGDATA. VM snapshot сервера DS отсутствует.

Так что предложено поступить так:

1. Сначала попробовать восстановиться из резервной копии. Это может сработать, а может нет, поскольку файловый бэкап при работающей бд может не соответствовать целостности оригинала при восстановлении.

2. Если способ №1 не удастся, то pg\_resetxlog, start, dump. Но после применения pg\_resetxlog целостность данных не гарантируется на 100%. Перед выполнением обязательно сделать копию \$PGDATA.

В результате способ №2 позволил успешно запустить datastore.

datastore.log содержал главные сообщения, указывающие на коррапт:

```
2016-08-14 21:05:27.131 BST 2004 57b0cf07.7d4 1 :LOG: database system was interrupted while in recovery at 2016-08-14 20:00:23 BST
```

```
2016-08-14 21:07:18.065 BST 2004 57b0cf07.7d4 8 :FATAL: failed to re-find parent key in index "19308" for split pages 113916/114000
```

pg\_resetxlog сбросили информацию в БД об используемых логах, указав параметры что именно сбрасывать из БД используя pg\_controldata:

Latest checkpoint's NextXID: 871/29047

Latest checkpoint's NextOID: 127456

```
pg_resetxlog.exe -o 127456 -x 29047 -f "C:\datastore\data"
```

Transaction log reset

pg\_start запустили базу данных. Если Psql конектится успешно, сделать бекап:

```
pg_dump -v -host=PGHOST -port=PORT -c -Fc dpa.
```

Далее initdb, restore на новом протестированном кластере.

### 3. Datastore запущен и работает, но запросы к индексированным таблицам, обновляемым в период сбоя не выполнялись.

С помощью REINDEX перестроили существующие индексы. Non-concurrent REINDEX будет блокировать таблицы:

```
REINDEX TABLE node {FORCE}
```

```
REINDEX TABLE alias_list
```

```
REINDEX TABLE backupjob
```

```
REINDEX TABLE request_history
```

Выполнили vacuum full, чтобы освободить больше пространства:

```
VACUUM FULL VERBOSE node
```

```
VACUUM FULL VERBOSE alias_list
```

```
VACUUM FULL VERBOSE backupjob
```

```
VACUUM FULL VERBOSE request_history
```

Vacuum/analyze проанализировали статистику для главной таблицы node (slave+master):

```
VACUUM FULL ANALYZE VERBOSE node
```

\* Перед выполнением следует остановить app сервер, чтобы обеспечить исключительную блокировку таблицы.

Просмотреть план выполнения запроса:

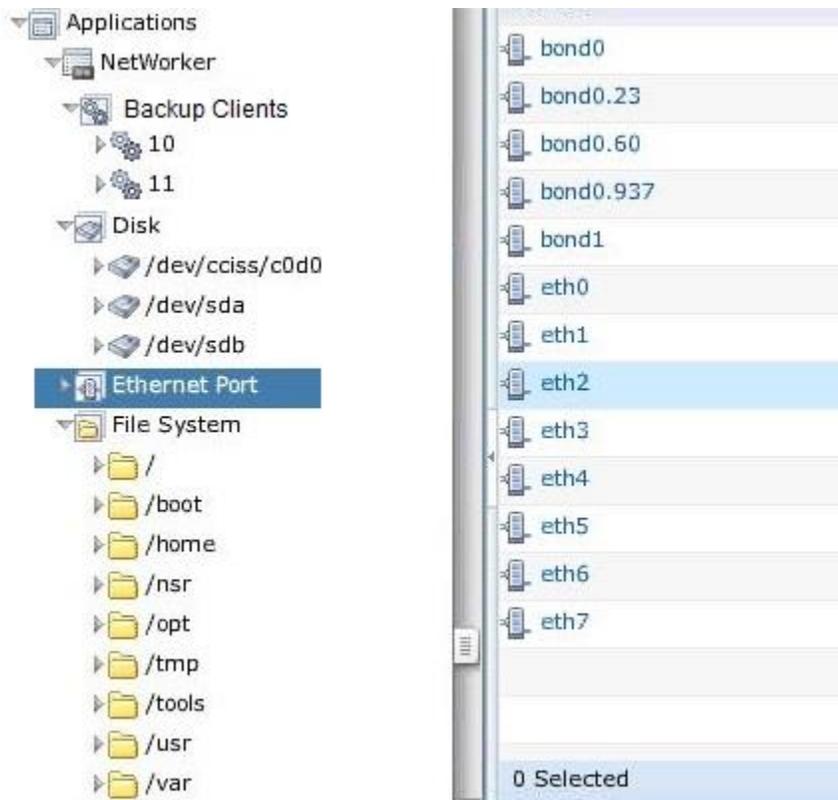
```
EXPLAIN ANALYZE WITH RECURSIVE q AS (SELECT n.*, 1 as level FROM node n  
WHERE f_partof='00000000-0000-0000-0000-000000000001') UNION ALL
```

```
SELECT n2.*, level+1 as level FROM q JOIN node n2 ON n2.f_partof=q.f_id) SELECT *  
FROM q
```

\* Обновление статистики и высвобождение пространства более релевантно для таблиц обновляемых чаще.

#### 4. App сервер запущен и работает. Устранение проблем Retention механизма.

DPA UI



При запуске репорта выбрасывалось исключение:

```
LOG 588128d4.2226:ERROR: update or delete on table "backupjob" violates foreign key constraint "backup_fkey" on table "error"
```

Запрос для получение информации о Foreign Key constraints:

```
SELECT tc.constraint_name, tc.constraint_type, tc.table_name, kcu.column_name, tc.is_deferrable, tc.initially_deferred, rc.match_option AS match_type, rc.update_rule AS on_update, rc.delete_rule AS on_delete, ccu.table_name AS references_table, ccu.column_name AS references_field FROM information_schema.table_constraints tc
```

```
LEFT JOIN information_schema.key_column_usage kcu ON tc.constraint_catalog =
kcu.constraint_catalog AND tc.constraint_schema = kcu.constraint_schema AND
tc.constraint_name = kcu.constraint_name
```

```
LEFT JOIN information_schema.referential_constraints rc ON tc.constraint_catalog =
rc.constraint_catalog AND tc.constraint_schema = rc.constraint_schema AND tc.constraint_name
= rc.constraint_name
```

```
LEFT JOIN information_schema.constraint_column_usage ccu ON rc.unique_constraint_catalog
= ccu.constraint_catalog AND rc.unique_constraint_schema = ccu.constraint_schema AND
rc.unique_constraint_name = ccu.constraint_name
```

```
WHERE tc.table_name = 'error'
```

```
AND tc.constraint_name = 'backup_fkey';
```

constraint_name	constraint_type	table_name	column_name	on_update	on_delete
backup_fkey	FOREIGN KEY	error	job_id	NO ACTION	NO ACTION

Реконструировать Foreign Key constraint:

```
ALTER TABLE error DROP CONSTRAINT backup_fkey;
```

```
ALTER TABLE error ADD CONSTRAINT backup_fkey FOREIGN KEY (job_id)
REFERENCES backup (f_id) MATCH SIMPLE ON UPDATE CASCADE ON DELETE
CASCADE;
```

## 5. Удаление дубликатов данных, которые появились вследствие восстановления.

Для нахождения дубликатов:

```
SELECT * FROM nodegraphlink o
WHERE exists ( SELECT 'x'
FROM nodegraphlink i
WHERE i.f_id = o.f_id
AND i.ctid > o.ctid
);
```

Для удаления дубликатов:

```
DELETE from nodegraphlink
WHERE exists ( select 'x'
FROM nodegraphlink i
WHERE i.f_id = nodegraphlink.f_id
AND i.ctid > nodegraphlink.ctid
);
```

Проверить уникальность алиасов, удаление дубликатов:

```
DELETE from aliases_list original USING aliases_list dupes WHERE
CAST(dupes.alias_node_id AS VARCHAR(36)) = CAST(original.alias_node_id AS
VARCHAR(36)) AND dupes.aliases = original.aliases AND dupes.aliases_order >
original.aliases_order
```

Данные в Request History наиболее критичны для клиента, так как истории запросов сохраняет всю информацию о джобах и утратив ее - невозможно восстановить. До момента выключения базы данных вся история для всех нод сохранилась без потерь.

### Job Summary

Completed	Succeeded	Failed	Active	Size (GB)	Success Rate ...
1,089	1,089	0	1	117	100
1,097	1,097	0	5	206	100
0	0	0	0	0	0
0	0	0	0	0	0
3	0	3	0	0	0
132	132	0	0	9	100
96	96	0	0	1	100
0	0	0	0	0	0
0	0	0	0	0	0
56	56	0	0	1	100

Мы наблюдали небольшие задержки и тайм-ауты при регистрации агентов нескольких NetWorker хостов, на которых произошел сбой:

```
<REQUESTTIMEOUT>10800</REQUESTTIMEOUT>
```

Несколько агентов пришлось перерегистрировать в DPA.

Вывести список имен и алиасов агента:

```
$ sudo ../app/dpaagent -i
```

## 6. Проблемы политики безопасности и администрирования.

SID- однозначно идентифицирует хост и связанную с ним информацию в домене. В виндовом домене проблем с безопасностью быть не должно, так как сиды и имена хостов будут всегда разные. На NetWorker серверах было обнаружено перетекание хост-неймов с одного хоста на другой.

Если произойдет переименование доменного имени, то это может вызывать проблемы с security policy для учетных записей (админов/не-админов).

Безопаснее выдавать права на время и отзываться их по его истечении.