Microsoft

Home > Exchange > Exchange Team Blog > Repairing Exchange databases with ESEUTIL - when and how?

Back to Blog | < Newer Article | Older Article >

The_Exchange_Team Microsoft

06-18-2004 1

# Repairing Exchange databases with ESEUTIL - when and how?

Here's a nightmare scenario for an Exchange administrator:

The disk system where your Exchange databases live goes insane and damages the database. No big deal, you think, I'll just restore it from backup and roll forward with the transaction logs. No data loss, although it will cost some downtime.

Then you find that somebody put a 2 liter bottle of soda on top of the backup tapes. A bottle with a leaky seam. Not all things go better with Coke!

Your Exchange database is unstartable and your backups are bad. What do you do next?

Exchange includes two very sophisticated utilities that will come to your rescue: Eseutil and Isinteg. If there is salvageable data in your injured database, they'll stitch it back together for you.

In actual practice, these two utilities are remarkably successful in fixing up a database (almost) like bran ators who've g backups because they know they can count on the repair tools to

Downloads | Exchang... | Get-RBA... | ExchActi... | database... | Health C... | Deployin... | Message Tracking Confi... | 100% | Clear (almost) always save their bacon.

That parenthetical (almost) is why you can't rely on repairing a database as your primary method of data recovery. Nothing beats having an extra copy of your data safely off in a second location (like on a backup tape). And repair can only repair what's still there. Massive damage to the existing database or complete loss of the drives is an all too frequent occurrence, even with today's redundant disk hardware.

Here's how to repair an Exchange database that won't start:

1. Make sure that the database really isn't startable. Sometimes, the database is fine, but something else is getting in the way. Here are a couple of suggestions for things to try before concluding your database must be repaired:

Check the Application Log--Exchange logs pretty detailed events for the startup of a database. Open any error events and use the Microsoft Knowledge Base (http://support.microsoft.com) to look up error numbers listed in the description fields. (One thing to remember about Exchange event logging: We have pretty generic event IDs for wide classes of events, and we usually put the real error or information in the description field. When searching the KnowledgeBase, use what's in the Description field, not just the Event ID.)

Restart the server. I know everyone is tired of hearing this advice from Microsoft support people, but let's face it--it's a really good way of clearing out random problems in an environment and getting you back to a good state when you don't have time to find the root cause before correcting the problem.

2. Make a copy of the database files(s) before you repair them. You'll probably skip doing this, but at least I told you. :)

If you're not sure where your database files are, or what they are called, you can find out in Exchange System Manager by accessing the database properties. The Database page lists the paths and names.

3. Verify that you have sufficient disk space to do the repair. As a general rule of thumb, you should have the equivalent of 20% of the database size. If you don't have that much free space on the drive where the database files are, you can use command line switches to redirect the temporary files created during repair to a different drive.

4. Run Eseutil in /P (repair) mode.

The easie ve both database files (.EDB and .STM) in the same directory (which they usually are). If they're in different places, you're going to have to point to the files on the command line.

Eseutil is found in the \exchsrvr\bin directory created when you install Exchange on a server. You may want to add \exchsrvr\bin to your system path for convenience.

Here is a loaded up Eseutil repair command line:

```
Eseutil /P c:\exchsrvr\mdbdata\DB1.EDB /Sd:\exchsrvr\mdbdata\DB1.STM /Te:\TEMPREPAIR.EDB
```

This command line will repair DB1.EDB located on C: along with its matching .STM file located on D: and will put the temporary file on the E: drive.

If your streaming database file (.STM) is not matched to the database file (.EDB) or it has a problem that is blocking repair, you can repair without it by adding the /createstm switch to the repair command line. This will destroy the .STM file and repair only the data in the .EDB file. What do you lose if you lose the .STM file?

It depends on what kind of clients attach to your Exchange server. If everybody uses Outlook (MAPI protocol), then there will be very little user data in the .STM file. You may lose some in transit messages that haven't been delivered yet. If clients connect via POP3 or IMAP then most of their stuff will be in the .STM file, and its loss will be catastrophic to them. If clients use Outlook Web Access, messages will be in the .EDB file, but attachments sent will be in the .STM file.

Repair can take a while--hours. When it finishes, it will leave you with a very detailed log file of what it did called <database>.integ.raw.

You're not finished yet, however. There are two more steps to complete.

5. Run Eseutil in /D (defragment) mode.

Repair may leave index and space allocation problems in the database. Along with compacting the physical size of the file as much as possible, defragmentation also throws away and rebuilds the indexes and space trees (structures that track space in the database).

To defragment the database, you need space equivalent to the compacted size of the database, plus 10% for good luck. Microsoft PSS will use a rule of thumb that you need space equivalent to the original size of the database plus 10%, since they don't want to guess wrong about the ultimate shrinkage of the database.

But you space you'll need for database file) by running Eseutil /MS to do a "space dump" of the database.

At the top of the output, you will see a section labeled SLV Space Dump. Look for the Free total and multiply that by 4096. That's approximately how many bytes you can expect the .STM file to shrink by from defragmenting it. At the bottom of the output in the lower right corner, you will see another total number. Multiply that by 4096, and that's approximately how many bytes the .EDB file will shrink by. Subtract those numbers from the sizes of the database files, add 10% and that's how much space you really need to do a defragmentation.

As with repair, you can redirect the temporary file to a different drive if necessary, but that is going to cost you significant time. The way defrag works is that it creates a brand new database, and pumps all the old data into it. At the end of the process, the new database is copied back over the old one. If both are already on the same drive, this takes a split second. If on different drives, it takes however long it takes you to copy an X gigabyte file between your drives.

6. Run Isinteg in -fix -test alltests mode.

Isinteg is the only repair utility that understands the Exchange database as an Exchange database. That statement probably deserves some explaining.

ESE is a generic database engine that can be used by different applications, with Exchange happening to be one of them. Active Directory is another. Eseutil looks at the database as just another ESE database, and sees it's contents as a bunch of tables and indexes. It doesn't know or care whether this table holds a mail folder or that table has attachments in it. It just fixes up the tables so they are valid ESE tables once again. Eseutil doesn't understand that this database holds folders and messages--it just has tables and records.

Isinteg understands the relationships between those tables and records that turn them into folders and messages. If Eseutil had to delete a record that was a message, Isinteg knows how to decrement the count of messages for every folder that had a copy of that message. If you don't run Isinteg, clients will likely see strange things--like message counts that are off, messages that appear in the Inbox but can't be read, and so on.

When you run Eseutil, you can move database files to temporary locations to make repairs. But to run Isinteg, you must put the database back in the location from which it is normally mounted. The reason for this is that Isinteg actually mounts the database in order to read it through the Information Store process, while Eseutil reads databases as raw files.

At the end of an Isinteg fix run, you will likely see hundreds to thousands of warnings. This is normal. Isinteg was originally created as an internal test utility and its output is quite verbose, deliberat d of a successfu d be zero errors reported. If there is even one error, you should run Isinteg again.

If successive runs of Isinteg do not decrease the number of errors reported, and you cannot get the error count down to zero, then you should not rely on this database in production. You should move mailboxes from it or otherwise salvage data, and then discard it. It's relatively infrequent for Isinteg to not be able to get the error count down to zero, however.

So, that's the basics of Exchange database repair:

You run Eseutil /P first.
Then you run Eseutil /D.
Then run Isinteg -fix -test alltests.

This is no substitute for making good backups and is likely to take considerably longer to accomplish than restoring from backup. As a ballpark estimate, expect to spend an hour per gigabyte of data to get through the whole repair process.

Is your database guaranteed to be OK after this regimen? No, not 100%. But it very likely is. Over the years, these utilities have been improved and tweaked repeatedly to handle new problems, and they are now pretty comprehensive. Nonetheless, we still get the occasional curve ball thrown at us and find something that the utilities can't handle, and new improvements are made.

Whether you should leave a repaired database in production is a matter of philosophical disagreement, with your own tolerance for risk factoring in. If you want to be 100% sure that the database is completely OK after successful repair, rather than just being 99.9% sure, then I suggest moving all mailboxes to another database, and then deleting the repaired database. After deletion, next time you start the database, a new one will be automatically generated, and you can move the mailboxes back. If it's a public folder database, then replicate all folders, delete the database, and replicate them all back to a fresh database.

Mike Lee

Tags: administration All Posts Storage troubleshooting

0 Likes

Share

## 8 Comments

Not applicable 06-18-2004 07:

I'd have to agree, the two utilities are remarkably successful.

I've had nothing but mediocre success with repair utils in the past. Even with SQL Server, repair results could be a lot more desirable.

My Exchange2k databases corrupted for some reason last year and I didn't have a recent backup (its a bit of an experimental machine with only 3 users). After diagnosing the problem using Event Viwer, I was directed by online resources to use the repair utility. I honestly did not expect it to work. Not only did it work, but it worked exceptionally well. The repair was fast and no data was lost.

My Exchange database repair experience by far the best I've encountered.

0 Likes

Not applicable 06-18-2004 07:

You need to have your Testing & Learning folks read this. This is not what they say to do on the Exchange exams.

0 Likes

Not applicable 06-21-2004 05:

This is the exact steps we follow in our enviroment . There is lot of Documents out there in MS site for ESEUTIL with details.

But you have given simple steps for a novice ..

0 Likes

06-29-2004 10:

Not applicable

very good article Mike, will certainly help alot of administrator sort out their exchange rubuild problems.

0 Likes

07-12-2004 08:

Not applicable

Thank you Mike... You saved the day.

0 Likes

07-22-2004 09:

Not applicable

Well well well, you DO know things.
Thankyou, you have saved MY day.

0 Likes

03-04-2005 01:

Not applicable

davids brain blog &raquo; Blog Archive &raquo; today a blog saved my

Downloads | Exchang... | Get-RBA... | ExchActi... | database... | Health C... | Deployin... | Message Tracking Confi... | 100% | Clear

0 Likes

Not applicable

02-17-2007 02:

PingBack from http://jarchitect.info/blog/?p=14

0 Likes

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

Comment

What's New

Surface Pro X

Surface Laptop 3

Surface Pro 7

Windows 10 Apps

Office apps

Microsoft Store

Account profile

Download Center

Microsoft Store support

Returns

Order tracking

Store locations

Buy Online, pick up in store

In-store events

Education

Microsoft in education

Office for students

Office for schools

Deals for students & parents

Microsoft Azure in education

Downloads Exchang… Get-RBA… ExchActi… database… Health C… Deployin… Message Tracking Confi… 100% Clear

Enterprise

Azure

AppSource

Automotive

Government

Healthcare

Manufacturing

Financial Servcies

Retail

Developer

Microsoft Visual Studio

Window Dev Center

Developer Network

TechNet

Microsoft developer program

Channel 9

Office Dev Center

Microsoft Garage

Company

Careers

About Microsoft

Company news

Privacy at Microsoft

Investors

Diversity and inclusion

Accessibility

Security

English (United States)

Sitemap Contact Microsoft Privacy & cookies Terms of use Trademarks Safety & eco

About our ads

Downloads Exchang… Get-RBA… ExchActi… database… Health C… Deployin… Message Tracking Confi… 100% Clear