記事
Toshihiko Minamoto · 2021年6月21日 9m read

SYSLOG - その正体と意味するもの

この記事では、syslogテーブルについて説明したいと思います。 syslogとは何か、どのように確認するのか、実際のエントリはどのようなものか、そしてなぜそれが重要であるのかについて説明します。
syslogテーブルには、重要な診断情報が含まれることがあります。 システムに何らかの問題が生じている場合に、このテーブルの確認方法とどのような情報が含まれているのかを理解しておくことが重要です。

syslogテーブルとは？

Caché は、共有メモリのごく一部を使って、関心のある項目をログに記録しています。
このテーブルは、次のようなさまざまな名前で呼ばれています。

- Caché システムエラーログ
- errlog
- SYSLOG
- syslogテーブル

この記事では、単に「syslogテーブル」と呼ぶことにします。

syslogテーブルのサイズは構成です。 デフォルトは500エントリで、10～10,000エントリの範囲で構成します。 syslogテーブルのサイズを変更するには、システム管理ポータル -> システム管理 -> 構成 -> 追加設定 -> 詳細メモリに移動し、「errlog」行の「編集」を選択します。
そこにsyslogテーブルに必要なエントリの数を入力してください。

syslogテーブルのサイズを変更する理由は？

syslogテーブルが500エントリに構成されている、501番目のエントリによって最初のエントリが上書きされ、そのエントリの情報は失われてしまいます。
これはメモリ内にあるテーブルであるため、出力を明示的に保存しない限りどこにも永続されません。 また、Caché を停止した場合には、以下に説明する方法でエントリをconsole.logファイルに保存するように構成していない限り、すべてのエントリは失われてしまいます。

Caché が多くのエントリをsyslogテーブルに書き込んでおり、問題の診断目的でそのエントリを確認する場合、テーブルのサイズが十分に大きくなければ、エントリは失われてしまいます。
syslogテーブルの **Date/Time** 列を見ると、テーブルが書き込まれた期間を判別できます。
その上で、どれくらいのエントリ数を設定するのかを決めることができます。
個人的には、エントリを失わない数で判断を誤る方が良いと思います。
このことについては、以下でより詳しく説明します。

syslogテーブルの確認方法

syslogテーブルの確認方法にはいくつかあります。

1. Caché ターミナルプロンプトから、%SYSネームスペースで「Do ^SYSLOG」を実行する。
2. Caché ターミナルプロンプトから、%SYSネームスペースで「do ^Buttons」を実行する。
3. 管理ポータル -> システム操作 -> 診断レポートに移動する。

4. -e1オプションでcstatを実行する。
5. Cachehungを実行する。
6. シャットダウン中にsyslogテーブルをconsole.logファイルに書き込むようにCaché を設定し、console.logファイルを確認する。 設定するには、システム管理ポータル -> 構成 -> 追加設定 -> 互換性移動し、「ShutDownLogErros」行で「編集」を選択します。 Cachéのシャットダウン中にsyslogのコンテンツをconsole.logに保存する場合は「true」、保存しない場合は「false」を選択します。

syslogのエントリは？

syslogテーブルの例を以下に示します。
この例は、Cachéターミナルプロンプトで「^SYSLOG」を実行して得られたものです。

```
%SYS>d ^SYSLOG

Device:
Right margin: 80 =>
Show detail? No => No
Cache System Error Log printed on Nov 02 2016 at  4:29 PM
------------------------------------------------------------
Printing the last 8 entries out of 8 total occurrences.
Err  Process         Date/Time                              Mod     Line      Routine
                            Namespace
9     41681038     11/02/2016 04:44:51PM     93        5690     systest+3^systest
     %SYS
9     41681038     11/02/2016 04:43:34PM     93        5690     systest+3^systest
     %SYS
9     41681038     11/02/2016 04:42:06PM     93        5690     systest+3^systest
     %SYS
9     41681038     11/02/2016 04:41:21PM     93        5690     systest+3^systest
     %SYS
9     41681038     11/02/2016 04:39:29PM     93        5690     systest+3^systest
     %SYS
9     41681036     11/02/2016 04:38:26PM     93        5690     systest+3^systest
     %SYS
9     41681036     11/02/2016 04:36:57PM     93        5690     systest+3^systest
     %SYS
9     41681036     11/02/2016 04:29:45PM     93        5690     systest+3^systest
     %SYS
```

列の見出しを見ればエントリのどの項目が何であるかは明確のようですが、それらについて説明します。

Printing the last 8 entries out of 8 total occurrences

これは、syslogテーブルエントリの一部ではありませんが、確認することは重要なので、ここで説明します。
この行から、いくつのエントリがsyslogテーブルに書き込まれたかがわかります。
この例では、Cachéが起動してから8つのエントリしか書き込まれていません。
エントリ数が少ないのは、これが私のテストシステムであるためです。「Printing the last 500 entries out of 11,531 total occurrences（発生総数11,531件中最新の500件のエントリを出力）」と書かれていれば、多数のエントリが見逃されていることがわかります。 見逃されたエントリを確認する場合は、テーブルサイズは最大10,000に増やすか、より頻繁にSYSLOGを実行してください。

Err

これは、関心のあるイベントについてログに記録される情報です。 エラーは必ずOSレベルで/usr/include/errno.h（Unix）から発生しているものと思われがちですが、実際には「必ず」ではなく、ほとんどの場合です。
ログに記録できるものであれば、何でも記録されます。 たとえば、診断アドホックのデバッグ情報、C整数の値、エラーコードの定義（10000を超えるもの）などを記録することができます。

どのように区別すればよいのでしょうか。
実際に、エントリの **Mod** と **Line** に示される、Cコードの行を確認する必要があります。
つまり、InterSystemsに連絡を取らなければ、それが実際に何であるかを区別することはできません。
では、わざわざ確認する必要はあるのでしょうか。 **err**
の意味を正確に知らずとも、ほかの情報を見ることで把握できることがたくさんあるからです。 エントリがたくさんある場合や、普段から目にするエントリとは異なるエントリがある場合には、InterSystemsに問い合わせることもできます。
syslogテーブルのエントリは、必ずしもエラー状態を示すものではないことに 注 意してください。

Process

これは、syslogテーブルにエントリを書き込んだプロセスのプロセスIDです。 たとえば、スタックしたプロセス、スピンし続けるプロセス、またはデッドプロセスがある場合、syslogテーブルに何か記録されていないかを確認できます。 記録されていれば、プロセスで障害が起きた理由の重要な手がかりになるかもしれません。

Date/Time

これは、エントリが書き込まれた日時です。 問題になった原因の手がかりを得るために、エントリの日時とシステムイベントの日時を相関 することは非常に重要です。

Mod/Line

Modは特定のCファイルに対応しており、Lineは、そのエントリをsyslogテーブルに書き込んだファイルの行番号です。 カーネルコードにアクセスできるInterSystemsの従業員のみが、これを検索できます。
このコードを調べるだけで、エントリに何が記録されたのかを正確に知ることができます。

Routine

syslogテーブルにエントリが書き込まれたときにプロセスが実行していたタグ、オフセット、およびルーチンです。 何が起こっているのかを理解する上で非常に役立ちます。

Namespace

これは、プロセスが実行していたネームスペースです。

では、err 9がsyslogテーブルに書き込まれた理由をどのようにして知ることができますか?

まず、示されているルーチンを確認します。 私の^systestルーチンは次のようになっています。

```
systest  ;test for syslog post
         s file="/home/testfile"
         o file:10
         u file w "hello world"
         c file
         q
```

syslogエントリでは、エントリが書き込まれた時に実行していたものはsystest+3だった と示されています。
この行は次のようになっています。

```
u file w "hello world"
```

プロセスがファイルに書き込もうとしていたため、これは実際にOSレベルのエラーである可能性があります。そこで、/usr/include/errno.hで9を探すと、次のようになっています。

```
#define EBADF   9       /* Bad file descriptor                  */
```

9はファイル関連であり、示されたコードの行はファイルに書き込もうとしていたため、これは実際にOSエラーコードであると推測するのが合理的です。

何がエラーになっているのかわかりますか?

これを解決するために、まず、/homeディレクトリとtestfileファイルの権限を確認しました。
両方とも777となっていたため、ファイルを開いて書き込むのは可能なはずです。
そこでコードをよく見ると、エラーに気づきました。 10秒のタイムアウトの前にコロンを2つ付けていなかったこと、そしてOpenコマンドにはパラメータを使用していなかったことに気づいたのです。
以下は更新したルーチンです。実際にエラーなしで終了、ファイルに書き込みます。

```
systest  ;test for syslog post
         s file="/scratch1/yrockstr/systest/testfile"
         o file:"WNSE":10
         u file w "hello world"
         c file
         q
```

最後に

syslogテーブルは、正しく使用すればデバッグに役立つ貴重なツールです。
使用するときは、次の点に 注 意してください。

1. **err**
   は必ずしもオペレーティングシステムのエラーとは限りません。ログに記録できるものは何でも記録されます。 記録された内容については、InterSystemsにお問い合わせください。
2. ログに記録されているその他の情報を使用して、何が起きているのかを判断します。 エラーと組み合わされたCOSコードの行から、それがOSのエラーであるかどうかについて、合理的に推測することができます。
3. 解決できない問題がある場合は、syslogテーブルを確認しましょう。手がかりが見つかるかもしれません。
4. **Date/Time**
   、エントリ数、および合計発生数を使って、syslogテーブルのサイズを増やす必要があるかどうかを判断します。
5. システムがsyslogテーブルに何を記録しているのかを把握しておきましょう。エントリに何らかの変更があったり、新しいエントリや異なるエントリが記録されたことに気づくことができます。
6. syslogテーブルのエントリは、必ずしも問題を示すものではありません。

#ターミナル #ヒントとコツ #ベストプラクティス #監視 #Caché

ソースURL: https://jp.community.intersystems.com/post/syslog-%E3%81%9D%E3%81%AE%E6%AD%A3%E4%BD%93%E3%81%A8%E6%84%8F%E5%91%B3%E3%81%99%E3%82%8B%E3%82%82%E3%81%AE