Linux Security 64 システムリソース/チューニングについて

Linux Security 64(LS64)スキャンアーキテクチャはエンドポイント内ファイルを素早く/漏れなくスキャンする為、システムリソースを最大限使用する設計となります。LS64 にはスキャン対象ファイルに対するファイルサイズ制限やメモリ/CPU 使用制限はなく、スキャン対象ファイルが巨大であったり、同タイミングで大量スキャンリクエストを発生させた場合(バックアップタスク等)、システムリソース消費が瞬間的に増加する事象が発生いたします。これはセキュリティを考慮した設計によるものとなり異常動作ではございません。下記に対処方法や、LS64 のシステム異常調査手法をご紹介させていただきます。なお、LS64 動作異常調査依頼は「スキャンができない/エラーが発生する」等の具体的事象以外、機能向上や特定環境におけるスキャンエンジンチューニングリクエストと判断される場合があります。その場合製品への反映には時間が掛かる可能性がございますため、早期解決を目指す場合は負荷低減対策/再インストール等の手法もご検討ください。

1. 【事象】
   CPU/メモリ/HDD アクセスが瞬間的に高騰する。
   単純にスキャンリクエストが多数発生し、リソースを最大限まで利用している可能性がございます。まず下記のスキャン負荷低減を図ってください。

   【負荷低減対策】
   - アーカイブスキャン無効化/階層低減
     アーカイブファイルは解凍時にリアルタイムスキャンで再度チェックされます。

   - スキャン除外
     access.log でスキャン時間(“Duration”)が長いファイルを確認し、スキャンから除外する。
     /var/opt/f-secure/baseguard/fsicapd/log/access.log
     ※主にデータベースファイルやバックアップデータ等の日常的に使われていないファイルが対象です。

     #CSV 変換方法
     Access.log 配置ディレクトリで下記コマンド実行
     cat access.log | jq -r '[.duration,.date,.file_path]|@csv' > CSVData

   - スケジュールスキャンのタイミング調整
     スケジュールスキャンは強制終了の機能が無いため、長期化するとサーバの繁忙期に重複する可能性があります。スキャン開始時間を早める等の対策を取ってください。Policy Manager に送信されるスキャンレポートでスキャンの完了時間を確認できます。

   - スケジュールスキャン時のリアルタイムスキャン無効化
     スケジュールスキャンが営業時間外等に稼働している場合、リアルタイムスキャンを該当時間帯のみ無効化する等の対処をとります。

   - 搭載メモリ追加
     搭載メモリが単純に足りていない可能性を考慮し、追加を行う。

※Access.log を参照すると、ファイルスキャンに時間がかかっているファイルを特定する事が可能です。Access.log はしきい値を超えた場合、ローテートされ最大 10 世代保存されます。このしきい値は 2022 年 9 月現在変更できません。その為、大規模環境等で障害時の負荷要因ファイル特定が必要な場合、下記のコマンドで別の access.log を作成して下さい。

```
# tail -F access.log | grep -v '"user_agent":"fsaccd' --line-buffered
>>/tmp/access-test.log
```

#問題再現が終了した後、tail を終了してください。

2. LS64 プロセスメモリ使用量が上昇を続ける。
   弊社製品スキャンプロセスがメモリを専有している事を経過観測してください。負荷低減対処を実施いただき改善しない場合は、異常動作がが疑われます。該当プロセスの gcore メモリダンプファイルを”プライバシー情報を含む事”を了承の上提出ください。

【経過観測】

- 上昇幅の経過観測
  TOP コマンド定期実行でメモリ使用量推移を観測してください。

- プロセスの継続性確認
  同一スキャンプロセスが継続的メモリ確保をしているか？を確認してください。

【負荷低減対策】

- アーカイブスキャン無効化/階層低減
- スキャン除外
- スケジュールスキャンのタイミング調整
- スケジュールスキャン時のリアルタイムスキャン無効化
- 搭載メモリ追加

【TOP のプロセス観測について】
“top | grep fsicapd” と"pstree -p | grep fsicapd"によりメモリ使用率が(長期間)高いスキャンプロセスの PID を確認する。

例)TOP

```
[root@ip-192-168-149-10 ~]# top | grep fsicapd
      PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
1737  root      20   0  764032   4316   1568    S     2.3   0.2   2:10.33 fsicapd
```

```
20819 root      20   0  546404 339428   3424 S        1.0  16.9   0:08.53 fsicapd
20820 root      20   0  526816 318856   3184 S        0.7  15.9   0:04.14 fsicapd
20821 root      20   0  527416 319640   3276 S        0 .7 15.9   0:04.20 fsicapd
20822 root      20   0  525796 318024   3336 S        0.7  15.8   0:04.02 fsicapd
```

例) pstree

```
[root@ip-192-168-149-10 ~]# pstree -p | grep fsicapd
             |-fsicapd(1737)-+-fsicapd(1746)-+-fsicapd(20819)
             |               |               |-fsicapd(20820)
             |               |               |-fsicapd(20821)
             |               |               `-fsicapd(20822)
             |               |-{fsicapd}(1755)
```

例)TOP を日時付加で記録するコマンド

```
[root@ip-192-168-149-10 ~]# # while true ; do top -b -n 1 | grep fsicapd; date ;
sleep 10; done > TopResult.txt
```

※ctrl + C で停止してください。

【gcore メモリダンプファイルの生成について】

“#gcore PID” によりプロセスメモリダンプを採取する。

例)

“#gcore 20819”

メモリダンプがカレントディレクトリに生成されます。

例)core.20819

パスワードを付加して暗号化します。(ZIP 等)

※これはダンプ内にプライバシー情報が含まれる可能性がある為です。

※メモリダンプ採取後プロセス操作(リスタート等)は不要です。

Tips

アーカイブスキャンのチューニング

LS64 では Linux Security 11.xx シリーズと比較し、アーカイブファイルスキャンに利用されるアルゴリズムが変更されております。その為 Linux Security11.xx とは異なるスキャン結果表示や、スキャン時間の変化が発生する可能性があります。これは、アルゴリズムの更新により、より深い階層のスキャンが可能となり、結果スキャンに必要な時間が増加する事に起因します。

LS11 シリーズでご利用いただけていたスキャン時間制限の調整等は、LS64 への搭載予定はありません。LS64 への移行後にスキャン時間増加が運用に影響を及ぼすレベルである場合、「アーカイブファイル内の脅威」に基づき、「アーカイブファイルのスキャンの無効化」、「アーカイブをネスティングレベルまでスキャン」を少なくする事でスキャン時間短縮が図れます。それでもスキャン時間短縮が計れない場合や、アーカイブスキャンの無効化が採用できない場合、稼働サーバ自体の機能向上をご検討ください。

Policy Manager で下記の箇所を変更してください。

アーカイブファイル内の脅威

アーカイブファイルをスキャンする際、たとえファイル内にウイルスを発見したとしても、解凍/駆除/再圧縮を行う事がシステム負荷上、現実的ではない為、f-secure 製品では脅威が検知されても駆除は行われません。F-Secure 製品は、この点について該当ファイルが解凍された際、リアルタイムスキャンがファイルをスキャンする事により安全性を確保しています。

スキャン除外によるチューニング

明らかに安全と思われるフォルダや、巨大なファイル(データベース)などについてはスキャンから除外する事でスキャン時間の短縮を図る事が可能です。

Policy Manager で下記の箇所を変更してください。

特定ファイルのスキャンにかかる時間の調査方法

Example:---

[root@cent08 ~]# time /opt/f-secure/linuxsecurity/bin/fsanalyze fspm-15.11.94088.exe

Engine versions: F-Secure Corporation Aquarius/18.0.687/2020-10-09_05 F-Secure Corporation Hydra/6.0.235/2020-10-09_02 F-Secure Corporation FMLib/17.0.607.475 (cf1875a)/2020-04-07_01 fsicapd/2.0.114

1 files scanned

real 0m5.177s

user 0m0.000s

sys 0m0.004s

セキュリティクラウド使用/不使用によるチューニング(リアルタイムスキャン/マニュアルスキャン)

上記に加え、LS64 はクラウドデータベースを常に参照し、パターンファイル更新よりも最新の脅威情報をスキャンに取り込みます。その為、一部ネットワーク(※)においてはこのインターネット接続がスキャンの遅延を招く事があります。マニュアルスキャン実行中以外の時間帯において LS64 の稼働率が高い場合、この機能を無効化する事で負荷を低減させる事ができます。

※高速ネットワークにおいては有効化をする事で逆にスキャン速度が向上する場合もあります。

Policy Manager で下記の箇所を変更してください。(この設定はリアルタイムスキャンカテゴリに位置しますが、マニュアルスキャンにも影響します。)

パターンファイル更新タイミングの調整

LS64 はパターンファイル更新時間を短縮するため、多くのコンピュータリソースを利用して更新適用作業を実行します。その為、稼働サーバのピークタイムとパターンファイル更新適用タイミングが重なる場合、稼働サービスに影響が出る事が考えられます。Policy Manager でパターンファイルの適用適用タイミングをコントロールする事でこの事象を回避可能です。(当設定はパターンファイルに含まれる”コンポーネントアップデート”に対してのみ有効です。ウイルス定義等の低負荷でセキュリティ上必須の更新については即座に適用されます。)こちらの設定を行っていても、コンポーネントアップデートのリリース後、リリースの有効期限に達すると強制的に適用が行われます。

また、パターンファイルは約 1 週間以上(パターンファイルのリリース状況により可変)更新が無いと、次回更新で完全パターン更新を行います。普段は差分パターン更新である為、完全パターン更新が発生すると更新負荷が更に高くなります。パターンファイルは高頻度で更新されている為、なるべく短い間隔での更新をおすすめいたします。

スキャンエンジンの更新によるパフォーマンス向上

LS64 はパターンファイル配信チャネルを通じて配信される、エンジン更新によりパフォーマンスの向上を図っております。LS64 の機能でこれらのエンジン更新を特定 Ver に固定する事ができますが、エンジン更新によるパフォーマンス向上の恩恵を受けられない事態が予想されます。

最新の LS64 バージョンリリース情報

https://community.f-secure.com/en/discussion/117646/linux-security-64-change-log/