EPICS アクセス・セキュリティ

Martin R. Kraimer
日本語翻訳：山本昇

1999 / 10 / 21

概要

EPICS　アクセス・セキュリティ機能についての解説。EPICS Input/Output Controller(IOC) Application Developer's GUid 第５章Access Securityからの抜粋。

1　概説

この節では、EPICSのアクセス・セキュリティ,すなわちIOCへのアクセスを制限するシステムについて解説する。この節は以下の部分から構成される。

1. Overviewこの節。
2. Quick startアクセス・セキュリティ機能を有効とするために必要な作業。
3. User's Guide アクセス・セキュリティの解説と使用法
4. Design Summary アクセス・セキュリティの要求仕様と設計概要
5. Application Programmer's Interface
6. Database Access Security EPICS IOC databaseのアクセス・セキュリティ機能
7. Channel Access Security EPICSチャンネルアクセスのアクセス・セキュリティ機能
8. Implementation Overview実装の概要

アクセスセキュリティ機能の要求仕様1992年にAN/APSでまとめられた。その要求仕様は、

`EPICS: Channel Access Security - Functional Requirements, Ned D. Arnold, 03/-9/92.`

として文書化されている。この文章はEPICS WWWで入手可能である。

2 Quick Start

あるIOC上で、アクセス・セキュリティ機能を有効とするためには次の手順が必要である。

- アクセス・セキュリティ・ファイルを作成する.
- IOC に必要な変更を加える。
  - レコード・インスタンスはASGフィールドに値を割り当ててもよい。空文字列が割り当てられた場合にはASGに”DEFAULT”が指定されたものと見なす。
  - アクセス・セキュリティ・ファイルは

    `asSubInit`

    および

    `asSubProcess`

    に関連付けされたサブ・ルーチン・レコードを用いてiocInitの後で再ロードすることができる。このレコードに値１を書き込むことで、再ロードが実行される。
- vxWorks のスタートアップ・スクリプトはiocInitが実行される以前に次ぎのコマンドを含んでいなければならない。.

  ```
  asSetFilename(<AccessSecurityFile>)
  ```

  次のコマンドは省略可能である。

  ```
  asSetSubstitutions(var1=sub1,var2=sub2,...))
  ```

以下の条件によってIOC上でのアクセス・セキュリティ機能の有効/無効が決定される。

- `asSetFIlename`

  がiocInitに先立って実行されなければ、アクセス・セキュリティ機能は有効とはならない。
- asSetFileが与えられ、accessセキュリティの最初の初期化が失敗すると、すべてのIOCへのアクセスは拒否される。
- アクセスセキュリティがいったん成功裏に初期化された後に、アクセスセキュリティを再初期化して実行しようとしてエラーが発生下歯合には、アクセス・セキュリティは失敗した再初期化の前の状態を保持する。

3 User's Guide

3.1 Features

アクセス・セキュリティは権限の無いチャンネル・アクセス・クライアントからIOCデータベースを守る。アクセス・セキュリティの権限は次の項目に基づいて決定される。

- Who: CA クライアントのUser ID.

- Where: ユーザのログインしているホストID. これはまた、CAクライアントが動作しているホストのHost IDでもある。したがって、ユーザがこのホストにローカルにログインしているか、あるいはリモートでログインしているかはチェックされない。
- What　レコードのそれぞれのフィールドが保護される。それぞれのレコードはASG(Access Security Group)フィールドを持つ。このフィールドにそれぞれのレコードが所属するASGが記録される。またそれぞれのフィールドはアクセス・セキュリティレベルを持つ。アクセス・セキュリティレベルは0または1である。それぞれのフィールドのセキュリティレベルはアスキーレコード定義ファイルで定義されている。従って同じレコード型のレコードでは同じフィールドは同じセキュリティレベルを持つ。
- When: アクセスルールはCALCレコードと同様のINPUTリンクと計算の機能を持つ。

3.2　制限

IOCデータベースはCAかVxWorks のシェルからのみアクセス可能である。IOCのローカルコンソールへのアクセスは物理的な保護がなされており、telnet/rloginによるアクセスは通常のUNIXのセキュリティおよび物理的なセキュリティにより保護されていると仮定している。高度に洗練された侵入に対しての防護は考慮されていない。IOCの所属するサブネットへのアクセスは通常の方法によって制限されるべきである。

3.3　定義

この文書では以下の用語を用いる:

- ASL: アクセス・セキュリティ・レベル(Access Security Level)
- ASG: アクセス・セキュリティ・グループ(Access Security Group)
- UAG: ユーザ・アクセス・グループ(User Access Group)
- HAG: ホスト・アクセス・グループ(Host Access Group)

3.4　Access Security Configuration File

この節では、アクセス・セキュリティ・グループ,ユーザ・アクセス・グループおよび ホスト・アクセス・グループの定義を記述するファイル(Access Security configuration file)の形式について述べる。IOCはアクセス・セキュリティ設定ファイルを読み込んでアクセス設定データベースを作成する。アクセス・セキュリティ設定ファイルの拡張子として普通.acfが使われる。まずは アクセス・セキュリティ設定ファイルの例を挙げる。

3.5 Simple Example

```
UAG(uag) {user1,user2}
HAG(hag) {host1,host2}
ASG(DEFAULT) {
     RULE(1,READ)
     RULE(1,WRITE) {
          UAG(uag)
          HAG(hag)
     }
}
```

この例では、任意のホスト上の任意のユーザに読み出しの許可を与え、host1およびhost2上のユーザuser1およびuser2にのみ書き込みの許可を与えるルールが設定されている。

3.6 アクセス・セキュリティ設定ファイルの形式

以下では以下の記述法を用いる。

- [ ] :省略可能な値のリスト
- | : 候補値のリストのセパレータ
- ... : 任意の個数の定義が与えられることを示す。
- # : #で始まる行はコメント行である。

一般的なアクセス・セキュリティ設定ファイルの形式は、

```
UAG(<name>) [{ <user> [, <user> ...] }]
...
HAG(<name>) [{ <host> [, <host> ...] }]
...
ASG(<name>) [{
   [INP<index>(<channel name>)
   ...]
   RULE(<level>,NONE | READ | WRITE) {
      [UAG(<name> [,<name> ...])]
      [HAG(<name> [,<name> ...])]
      CALC("<calculation>")
  }
...
}]
...
```

となる。

3.7 規則

UAG　ユーザ・アクセス・グループ(User Access Group). ユーザIDのリストである。リストは空でもよい。同じユーザIDが複数のUAGに繰り返し現れても良い。IOCについてはユーザIDはboot parameterのユーザフィールドからとられる。

HAG　ホスト・アクセス・グループ(Host Access Group). 計算機のホスト名のリストである。空であっても良い。同じホスト名が複数のHAGに繰り返し現れても良い。IOCについてはユーザIDはboot parameterのターゲットフィールドからとられる。

ASG　アクセス・セキュリティ・グループ(Access Security Group). ”DEFAULT”グループは特別な意味を持つ。メンバが空あるいはASG定義を持たないならば、そのメンバは”DEFAULT”グループに割当てられる。

INP<index> Index は”A”から”L”のうちの一文字をとる。これらはCALCレコードのINPフィールドと同じように取り扱われる。　ASG定義のRULE中でCALCフィールドが定義されているなら、INPフィールドが定義されていなければならない。

RULE　このフィールドがアクセス許可のルールを定義する。<level>は0あるいは1でなければならない。　レベル1の許可はレベル0での許可を意味する。許可の種類は、NONE, READ, WRITE のいずれかである。WRITE権限はREAD権限を意味する。標準的なEPICSレコードではVAL、CMD および RESフィールド以外はすべてレベル1にセットされている。

UAG アクセス権限を与えるユーザ・アクセス・グループのリストを指定する。UAGが与えられていない場合、すべてのユーザに権限を与える。

HAG　アクセス権限を与えるホスト・アクセス・グループのリストを指定する。UAGが与えられていない場合、すべてのホストに権限を与える。

CALC CALCレコードのCALCフィールドと同様の式を与える。ただし、結果はTRUE(1)あるいはFALSE(0)のいずれかで無ければならない。結果がTRUE(1)あるいはFALSE(0)であるかによって、このルールが適応されるか否かが決められる。実装では、結果は $0.99 < 1.01$で比較されている。

それぞれのEPICS レコードはASGフィールドを持っている。ASGフィールドにはそのレコードが所属するASGの名前が指定されている。そのレコードに接続されるCAクライアントのアクセス権限は以下の方法によって決定される。

1. レコードに指定されているASGの定義を探す。
2. ASG中のそれぞれのRULEが以下の順序で試される。
   (a) フィールドのレベルがRULEのレベル以下であるか。
   (b) UAGが定義されているなら、このユーザが指定されたUAGに含まれているか。UAGが空であればすべてのユーザが許可される。
   (c) HAGが定義されているなら、CAクライアントの動作しているHOSTが指定されたHAGに含まれているか。
   (d) CALCが定義されているなら、CALCの計算結果はTRUE(1)であるか。もしINP フィールドのいずれかがINVALIDアラーム状態にあるなら計算結果はFALSE(0)と解釈される。実装では、計算結果は $0.99 < 1.01$で比較されている。
3. ステップ2で得られた最大のアクセス権限が選択される。

上記手続は同一のRULE名、アクセスレベル、アクセス権限で複数の定義があることを許していることに注意しておく。

3.8 ascheck - アクセス・セキュリティ・設定ファイルの検証

アクセス・セキュリティ・設定ファイルを作成・変更したあと、このファイルの文法チェックをascheckコマンドで行うことができる。

```
ascheck < "filename"
```

これはUnixコマンドである。エラーはstdoutに表示される。エラ-が無ければなにも表示されない。このコマンドは文法エラーだけをチェックし論理的なエラーはチェックしないことに注意されたい。したがって、ascheckコマンドでエラーが無かったとしても、このファイルによるアクセス．セキュリティが問題を起こさないとは保証されない。

3.9 IOC アクセス・セキュリティの初期化

IOC上でアクセス・セキュリティ機能を有効にするためには、IOC初期化中　iocInitが実行される以前に、次のコマンドが実行されている必要がある。

```
asSetFilename( "< アクセス・セキュリティ・設定ファイル名>")
```

このコマンドが実行されていなければiocInit はアクセス・セキュリティ機能を有効としない。

　IOCがアクセス・セキュリティ機能を有効として初期化された後に、アクセス・セキュリティデータベースを変更することができる。次の節で説明する。サブルーチンレコードを用いた方法がよく使われる。vxWorksのシェルで

```
asInit
```

を実行することでもデータベースの変更を行える。また、asSetFilename コマンドをasInitコマンドの前に再度実行することができる。iocInitコマンド実行前にasInitコマンドを実行してはならない。

3.10 データベース設定

3.10.1 アクセス・セキュリティ・グループ

すべてのレコードには文字列を値としてもつASGフィールドがある。データベース設定ツールでこのフィールドに値を設定する。ASGフィールドが空白であるか、アクセス・セキュリティグループ・データベース内のどのASGとも等しくなければ、そのレコードは”DEFAULT”ASGに割当てられる。

3.10.2 サブルーチン・レコード・サポート

iocCore中に、サブルーチンレコードに接続可能な二つの関数、

```
asSubInit
asSubProcess
```

が定義されている。これらのルーチンが接続されたサブルーチンレコードを用意することで、IOCに新しいアクセスセキュリティ設定ファイルを再読み込みさせることが可能になる。アクセス・セキュリティ設定を変更するためには、

1. 最後に実行されたasSetFilenameコマンドに指定された、アクセスセキュリティ設定ファイルを新しい設定に変更する。

2. サブルーチンレコードのVALフィールドに1を書き込む。この書き込みはCA経由でもよい。

これによって、以下の事が実行される。

1. 値が1であると、asInitが呼ばれ、値を0に戻す。
2. レコードは非同期(asynchronous)処理が行われる。レコード処理は、新しいアクセス・セキュリティ設定が有効になったときか、タイムアウトが発生したところで完了する。　アクセス・セキュリティ設定に失敗した場合には、レコードはBRSVフィールドで決まるSeverity状態に置かれる。

3.10.3 Record Type 定義

レコードのそれぞれのフィールドはASL0かASL1のアクセス・セキュリティ・レベルを持っている。

4 Access Security コマンド

VxWorks shell上で使うAccess Security 関連のコマンドのリストである。

4.1 asSetFilename

Format: `asSetFilename ("<filename>")`

新しいアクセス・セキュリティ設定ファイルの名前を登録する。

4.2 asInit

Format: `asInit`

このコマンドはアクセス・セキュリティシステムを再初期化する。このコマンドはアクセス・セキュリティ設定ファイルを再度読み込んで、新しいアクセス・セキュリティ・データベースを構築する。このコマンドはアクセス・セキュリティ設定ファイルが`asSetFilename()`コマンドで変更された時や、アクセス・セキュリティ設定ファイルの内容そのものが変更された時に有効である。サブルーチンレコードを通じてこのコマンドを起動することができる。

4.3 asdbdump

Format: `asdbdump`

完全なアクセス・セキュリティ・データベースのダンプを出力する。

4.4 aspuag

Format: `aspuag ("<user access group>")`

ユーザ・アクセス・グループのメンバを出力する。ユーザ・アクセス・グループが指定されていない場合には、すべてのユーザ・アクセス・グループを表示する。

4.5 asphag

Format: `asphag ("<user access group>")`

ホスト・アクセス・グループのメンバを出力する。ホスト・アクセス・グループが指定されていない場合には、すべてのホスト・アクセス・グループを表示する。

4.6 asprules

Format: `asprules ("<access security group>")`

引数に指定されたアクセス・セキュリティ・グループのルールを出力する。アクセス・セキュリティ・グループが指定されていない場合には、すべてのアクセス・セキュリティ・グループについてのルールを表示する。

4.7 aspmem

Format: `aspmem ("<access security group>", <print clients>)`

引数に指定されたアクセス・セキュリティ・グループに属するすべてのメンバ(EPICSレコード)を出力する。アクセス・セキュリティ・グループが指定されていない場合には、すべてのアクセス・セキュリティ・グループについてのメンバを表示する。`<print clients>`が 1(0)であれば、それぞれのメンバに接続されたCAクライアントが表示される(されない)。