Armのキーテクノロジ TrustZoneセキュリティの研究

IoT注目マイコン Cortex-M23実験室

短期集中連載 第2回

TrustZoneセキュリティ・プログラムの基礎知識

中森 章

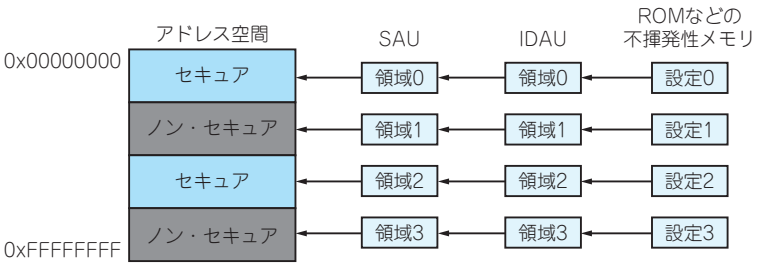

図1　セキュア/ノンセキュアなどのアドレス属性の設定メカニズム
セキュア，ノン・セキュア属性はSAUまたはIDAUで決定する（イメージ図）．メカニズム．SAUとIDAUの設定が異なる場合はIDAUの設定が優先される

Armv8-Mアーキテクチャ TrustZoneセキュリティの特徴

ここでは，Armv8-Mアーキテクチャでどのようにセキュリティを担保するかをまとめておきます．

①アドレス空間をセキュアとノン・セキュアに分割する
②プロセッサの実行状態をセキュアとノン・セキュアに分割する
③セキュア空間はセキュア状態，ノン・セキュア空間はノン・セキュア状態で実行する
④セキュア状態ではセキュア空間とノン・セキュア空間の資源をリード/ライトできる
⑤ノン・セキュア状態ではノン・セキュア空間の資源しかリード/ライトできない

以上が全てといっても過言ではありません．

Armv8-Mアーキテクチャの全体像は文献(1)を参考にできます．

● アドレス空間の分離

①のアドレス空間の分離はSAU（Security Attribution Unit）で行われます．Cortex-M23においてSAUは，通常，4または8の領域を持ち，それぞれのアドレス範囲とアドレス属性（セキュア，ノン・セキュア，ノン・セキュア・コーラブル）を決定します．しかし，SAUが存在しない場合や，SAUの設定を無視したい場合は，IDAU（Implementation Defined Attribution Unit）によって，SAUの設定を上書き（変更）できます．つまり，**図1**のような構成です．

● プロセッサ実行状態の移行

②のプロセッサの実行状態ですが，リセット後はセキュア状態です．ここから，ノン・セキュア状態に移行するためには，BXNS命令またはBLXNS命令を使用します．これは，普通の分岐命令と似ていますが，セキュア状態がノン・セキュアになる点が異なります．

しかし，ノン・セキュア状態からセキュア状態に移行するためには，厄介な作業が必要です．それは当然ですよね．ノン・セキュア状態からセキュア状態に簡単に移行できたらセキュリティなんてないも同然です．

セキュリティの属性として，セキュア，ノン・セキュアの他にノン・セキュア・コーラブル（ノン・セキュアから呼び出し可能の意）という属性があります．これは，ノン・セキュア状態からセキュア状態に移行する場合のセキュア・ゲート（関門ですね）を設けるための領域です．

ノン・セキュア状態からセキュア状態を呼び出すためには，ノン・セキュア・コーラブル領域に置かれたSG（Security Gate）命令をめがけて分岐します．SG命令を通過することでプロセッサの状態はセキュア状態になり，SG命令の直後に置かれた通常の（セキュ