M-036

映像監視システムにおけるセキュアなプロファイル設定機能の開発
Development of Secure Configuration Control Method for Video Surveillance System

若土 剛之† 阿倍 博信† 小林 信博† 中島 宏一†
Takayuki Wakatsuchi Hironobu Abe Nobuhiro Kobayashi Koichi Nakashima

1. 背景と狙い

社会不安の高まりから、映像監視市場が拡大し、映像監視システムに求める機能が多様化している。映像監視システムは、多様な機能を実現するために、入退室管理システム等の異種システムとネットワークで接続し、連携して動作するケースが増加している[1]。また、映像監視システム内部では、構成要素であるカメラ、レコーダ、ビューワ等が互いにネットワークで接続し、ネットワーク経由で映像配信を行っている。

上記のように様々な機器とネットワークで接続する映像監視システムは、セキュリティシステムとして高い信頼性が要求されるため、不正アクセスへの対策が必要である。構成要素のうち、特にカメラは UI がないために通常ネットワーク経由でプロファイル設定を行うことから、不正アクセスへの対策の必要性が高い。しかし、ネットワーク上の第三者によるカメラの設定変更等への対策は十分とは言えない。この場合の「プロファイル」とは、カメラの動作パラメータ情報（画像サイズ等）である。

本論文では、映像監視システムの構成要素であるカメラに着目し、上記のプロファイル設定における不正アクセスの脅威を除外する、暗号化・認証によるセキュア化の実現方式とその評価結果について述べる。

2. セキュアなプロファイル設定機能の開発

2.1 セキュア化方法

プロファイル設定機能をセキュア化するには、カメラへのパラメータ設定時に暗号化と認証を行い、許可されたユーザのみが設定・操作可能とする必要がある。暗号方式には、認証局を必要としない ID ベース暗号方式[2]を採用した。

2.2 IDベース暗号方式の概要と課題

ID ベース暗号方式は、PKI を利用した方式と比較して認証局が不要なため、LAN 環境の映像監視システムに低コストで導入できる。その反面、CPU の性能が低い組込み機器にとって処理負荷が高く、PC のように CPU の性能が高い機器でのみ使用されていた。すなわち、カメラで ID ベース暗号方式を使用するには、処理時間の短縮が課題である。

2.3 処理時間を短縮する方式

プロファイル設定に対し、処理内容によって初期設定と通常設定の 2 通りの設定動作を定義した。初期設定でのみ、ID ベース暗号方式を使用し、通常設定では高速な共通鍵暗号方式 MISTY[3]と HMAC[4]を使用する。通常設定で使用する暗号鍵は、機密性を高めるため、暗号鍵自体を送信して共有するのではなく、ユーザが操作する設定端末とカメラが個別に生成する鍵情報を初期設定時に送信し合い、2 つの鍵情報を足し合わせて生成し、共有する。

初期設定と通常設定の動作の詳細を以下で述べる。

†三菱電機（株）, Mitsubishi Electric Corporation

初期設定（1 回目）

(1) 設定するプロファイル・設定端末が生成した鍵情報・ID を送信（設定端末からカメラへ）
ID ベース暗号方式による暗号化と認証
(2) 通常設定で使用する暗号鍵を生成（カメラ側）
(3) プロファイル設定を更新
(4) 設定結果・カメラが生成した鍵情報・ID を送信（カメラから設定端末へ）
MISTY による暗号化（暗号鍵には設定端末が生成した鍵情報を使用）
(5) 通常設定で使用する暗号鍵を生成（設定端末側）

通常設定（2 回目以降）

<1> 設定するプロファイル・HMAC 鍵を送信（設定端末からカメラへ）
MISTY による暗号化（初期設定で生成した暗号鍵を使用）
<2> プロファイル設定を更新
<3> 設定結果・MAC 値を送信（カメラから設定端末へ）
HMAC による認証

初期設定と通常設定の動作シーケンスを以下の図 1 に示す。

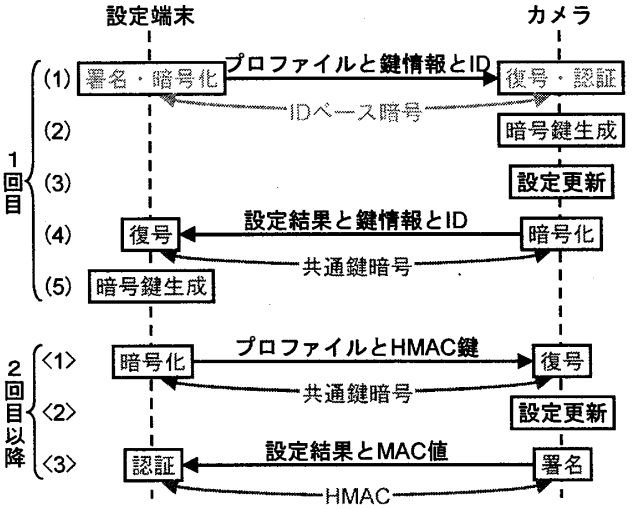

図 1．処理時間を短縮する動作シーケンス

上記のシーケンスで動作する初期設定と通常設定は、毎回 ID ベース暗号方式による暗号化と認証を行うプロファイル設定よりも処理時間が短くなることが自明である。設定端末は一般的に性能が高い CPU を搭載した PC を使用するため、プロファイル設定の処理時間はカメラの処理時間が大きな割合を占める。このため、以下ではカメラの処理時間に注目する。

カメラの処理時間の内訳を以下の図 2 に示す。

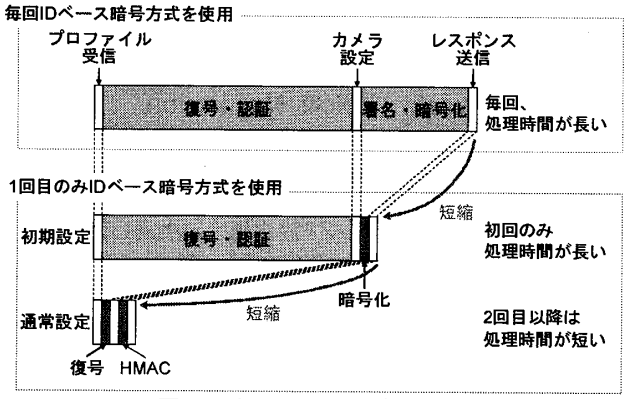

図 2．カメラの処理時間の内訳

2.4 カメラへの実装

セキュアなプロファイル設定は、UI 機能、送受信機能、カメラ設定機能、暗号化・復号機能によって実現される。このうち、カメラに実装する機能は、送受信機能、カメラ設定機能、暗号化・復号機能である。Web サーバ上で上記の 3 機能を実現する、2 つの CGI[5]（初期設定 CGI と通常設定 CGI）を開発した。

3. 性能評価

3.1 評価環境

今回開発した機能をカメラに適用するためのフィージビリティスタディとして、プロファイル設定の応答性能の評価を行った。一般的なカメラと同等の CPU 性能（ARM9 200MHz）を搭載したボード[6]（ターゲットノード）上に評価システムを構築した。初期設定 CGI と通常設定 CGI をターゲットノード上で動作させ、カメラの代わりに今回開発した暗号化・認証の機能を実現する。ターゲットノードが設定端末とカメラの間に入り、設定端末とターゲットノードの間は、今回開発したセキュアな通信を行い、ターゲットノードとカメラの間は、既存のプロトコルで通信を行った。

上記の評価環境の構成図を以下の図 3 に示す。

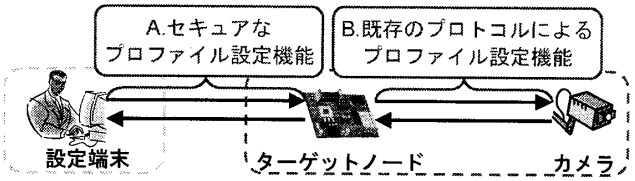

図 3．評価環境構成図

3.2 評価方法

初期設定 CGI と通常設定 CGI をターゲットノード上で動作させ、応答時間を評価した。評価システム上では、設定端末とターゲットノードの間（図 3 中の A）と、ターゲットノードとカメラの間（図 3 中の B）に通信があるが、ターゲットノードとカメラの間の応答時間を差し引いた値で評価した。なぜなら、セキュアなプロファイル設定機能をカメラに実装すると、ターゲットノードとカメラの間の通信がなくなるためである。

3.3 評価結果

初期設定 CGI と通常設定 CGI の応答時間を以下の表 1 に示す。

表 1．応答時間

初期設定	19.60 秒
通常設定	0.17 秒

この結果は、性能目標である通常設定 CGI の応答時間 1 秒以内（カメラ設定時に負担とならない時間）を達成した。

毎回 ID ベース暗号方式を使用する場合では、カメラが ID ベース暗号方式で署名と暗号化を行うため、初期設定よりも時間が長くなり、1 回のプロファイル設定に 19.60 秒以上かかる。

カメラの新規導入時の現地調整では、プロファイル設定を何度も変更し、パラメータを調整する。ここで、1 台のカメラに対して 10 回のプロファイル設定を行う場面を想定する。毎回 ID ベース暗号方式を使用した場合に 10 回のプロファイル設定でかかる合計時間は 196.00 秒以上であるのに対し、1 回目に初期設定、2 回目以降に通常設定を行った場合に 10 回のプロファイル設定でかかる合計時間は 21.13 秒である。初期設定と通常設定によるプロファイル設定は、毎回 ID ベース暗号方式を使用するプロファイル設定と比較して、カメラ 1 台当たりの設定に必要な時間を 174.87 秒以上短縮できる。

初期設定と通常設定によるプロファイル設定と、毎回 ID ベース暗号方式を使用するプロファイル設定のカメラ 1 台当たりの応答時間を以下の図 4 に示す。

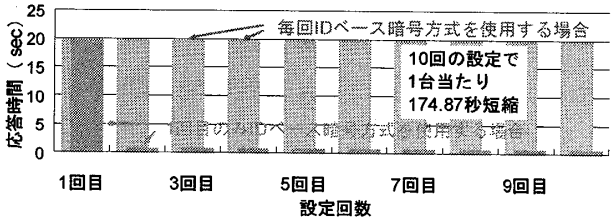

図 4．カメラ 1 台のプロファイル設定の応答時間

4. 結論

カメラのプロファイル設定機能に対し、ID ベース暗号方式を使用することにより、設定のセキュア化を実現した。ID ベース暗号方式を採用するにあたり、ID ベース暗号方式を使用する初期設定と、初期設定時に設定する暗号鍵を用いる共通鍵暗号方式を使用する通常設定の 2 通りの構成にすることにより、課題であった処理時間を短縮できた。

今後は、より実システムに近いプロトタイプを構築し、動作検証・評価を行う予定である。

参考文献

[1]富士経済, “2008 セキュリティ関連市場の将来展望”, (2008.5)
[2]G. Appenzeller, L. Martin, M. Schertler, RFC5408, “Identity-Based Encryption Architecture and Supporting Data Structures”, (2009.1)
[3]三菱電機, “About Misty”, (http://www.mitsubishielectric.co.jp/security/info/misty/about.html)
[4]H. Krawczyk, M. Bellare, R. Canetti, RFC2104, “HMAC: Keyed-Hashing for Message Authentication”, (1997.2)
[5]D. Robinson, K. Coar, RFC3875, “The Common Gateway Interface (CGI) Version 1.1”, (2004.10)
[6]アットマークテクノ, “Armadillo-9”, (http://www.atmark-techno.com/products/armadillo/a9)