M-086

鍵の自動取得機構を装備した IPsec 通信のモニタリングシステムの実装

中村元彦†，寺田真敏††，千葉雄司††，矢田健一††，土居範久†
Motohiko Nakamura, Masato Terada, Yuji Chiba, Kennichi Yada, Norihisa Doi

1. はじめに

今日，コンピュータネットワークは，情報システム基盤として重要な役割を担っている．コンピュータネットワークに障害が発生すると組織活動に支障をきたすため，ネットワーク管理者には迅速な障害対応が求められている．

障害を解決するには，まず，障害の原因を見つけ出す必要がある．その有効な手段の 1 つに，パケットアナライザを使用して，ネットワーク上を流れる IP パケットを捕捉し，その内容を解析する手法がある．解析の対象とする通信が平文通信の場合，パケットは暗号化されていないため，パケットアナライザを使用して，モニタリングとパケット解析をおこなうことで通信内容を把握できることが多い．暗号化通信の場合，解読に必要な鍵があれば，暗号化されたパケットの暗号化部分を解読して，同様に通信内容を把握できる．現在市場に出回っているパケットアナライザの中には，暗号化通信をモニタリングし解析する機能を持つ製品もあるが，操作が煩雑である．なぜなら，これらのパケットアナライザは解読に必要な鍵を手動で取得し設定する必要があり，また，暗号化通信の鍵更新時には鍵を適宜切り替えなければならないからである．

そこで，本研究では IPsec による暗号化通信を対象として，鍵の自動取得機構を装備した暗号化通信向けモニタリングシステムを実装した．実装したシステムの特徴は，つぎの通りである．

- **フィルター型の解読機能**
  暗号化通信の解析は，パケットの暗号の解読処理と可読な形式への変換処理の 2 つの処理からなる．これらの処理は，パケットアナライザのひとつである tcpdump の出力に対して，ネットワーク管理者の管理者端末（障害の解析，モニタリングをおこなう端末）上でフィルター型のソフトウェアとして動作する．このため，パケットアナライザの改造は必要なく，付加ツールとして暗号化通信をモニタリングできる．
- **鍵の自動取得機能**
  暗号化の解読に必要な鍵を IPsec クライアント（IPsec による暗号化通信をおこなうユーザの端末，本稿ではクライアントと略す）から自動的に取得し，管理者端末上で動作するフィルター型の暗号を解読するソフトウェアに鍵を設定する．また，クライアント間の Re-Key（IPsec における鍵更新）を検知し，新しい鍵を自動再取得する．

本稿では，上記機能を備えたモニタリングシステムの実装方法について述べる．つぎに，実装したシステムの評価として，解読機能の処理性能と鍵の自動取得機能が解読処理に与える影響について述べる．

2. 既存のパケットアナライザの動向

IPsec による暗号化通信をモニタリングすることが可能なパケットアナライザとして，tcpdump や NetCocoon Analyzer [1]がある．いずれも暗号化通信をモニタリングするためには，あらかじめ解読のための鍵を手動で取得しておき，モニタリング開始時に鍵を設定する必要がある．

鍵の取得方法は，IPsec の実装方法によってさまざまであり，セキュリティゲートウェイ（VPN 装置）のデバッグファイルや OS (Linux や BSD の場合）のコマンドを入力することで取得することが一般的である．鍵の入力方法はパケットアナライザにより異なり，tcpdump では起動時の-E オプションから，NetCocoon Analyzer では専用の入力画面から設定する．

3. モニタリングシステムの概要

本論文で提案する鍵の自動取得機能を装備した IPsec 通信のモニタリングシステムについて詳述する．

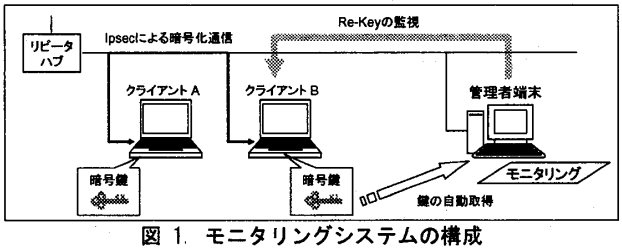

図 1. モニタリングシステムの構成

3.1 システム構成

本システムは，モニタリング対象となるクライアントと，モニタリングをおこなう管理者端末からなる（図 1）．また，リピータハブにクライアントおよび管理者端末を接続することによりクライアント間のパケットキャプチャを実現する．

3.2 ソフトウェア構成

鍵の自動取得機能（図 2 (a), (b)）

管理者端末上で，解読に用いる鍵の自動取得を実現するために，鍵取得モジュールと鍵送信モジュールを作成した．鍵送信モジュールは，クライアント上でデーモンプロセスとして動作し，管理者端末上で動作する鍵取得モジュールからの要求に応じて鍵を送信する．

フィルター型の解読機能（図 2 (c)）

管理者端末でのモニタリングは，tcpdump によるパケットキャプチャと，その出力を処理する解読モジュールからなる．また，パケット解読モジュールは，鍵の取得の際に鍵取得モジュールを呼び出す．

3.3 モニタリングの手順

本モニタリングシステムは，なんらかのネットワーク障害が発生した際に，ネットワーク管理者がモニタリング用システムを起動することを想定している．モニタリング用ソフトウェアの処理フローは，つぎのとおりである（図 2）．

1) 鍵取得モジュールは，クライアント上で動作している鍵送信モジュールとの間で TCP コネクションを確立する．コネクション確立後，鍵送信モジュールは鍵と暗号化アルゴリズムの情報を暗号化（モジュール間で使用する共通鍵は事前に設定しておく）して鍵取得モジュールに送信する．

† 中央大学 大学院 理工学研究科 情報工学専攻
†† 中央大学 研究開発機構

2) 鍵取得モジュールは取得した情報を復号し，解読モジュールに引き渡す．解読モジュールは，受け取った情報を使って tcpdump の出力を解読し，さらに，可読な形式に変換して出力する．出力先はターミナル画面か，ファイルのいずれかを選択できる．

3) 解読モジュールは，Re-Key に対応し，解読用の鍵切替えをおこなうため，tcpdump の出力内容を監視する．そして，Re-Key を検出したら，新しい鍵を再取得する．

・Re-Key の検出

Re-Key では IKE (Internet Key Exchange) パケットを使用し，暗号化通信は ESP (Encapsulating Security Payload) パケットを使用するため，このパケットのプロトコルの違いを利用して検出する．IKE では，現在の暗号化鍵の寿命が切れる前に，新しい鍵の生成のため事前に Re-Key をおこなうため，新しい鍵が反映されたパケットの検出が必要となる．

・解読に使用する鍵切り替えのタイミングの検出

新しい鍵で暗号化された最初のパケットを検出する方法は，ESP パケットのシーケンス番号を利用する．シーケンス番号は，Re-Key 時に"1"から新たに割り当てられるため，tcpdump の出力内容のシーケンス番号部分に"0x1"を検出した時点で，鍵切り替えをおこなう．

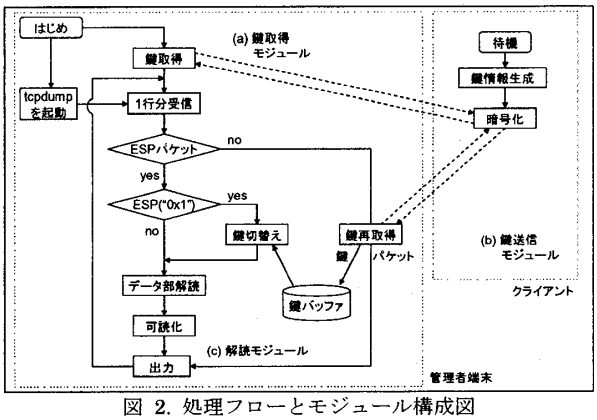

図 2. 処理フローとモジュール構成図

4. モニタリング処理と鍵の取得に関する評価

4.1 実験環境

実験に使用したハードウェア環境を表 1 に示す．

表 1. 実験環境

	管理者端末	クライアントA	クライアント B
CPU	Pentium4 2.8C	Pentium4 2.6C	Celeron 500[MHz]
Memory	1024 [Mbyte]	512 [Mbyte]	192 [Mbyte]
OS	Redhat Linux 9	FreeBSD 4.6.2	FreeBSD 4.6.2

つぎに，実験に利用したソフトウェアについて述べる．クライアント間で Re-Key を実現する IKE は，FreeBSD のパッケージ racoon を使用した．解読できるアルゴリズムは DES 限定で実装したため，racoon において暗号化アルゴリズムは静的に指定した．なお，パケット解読，鍵取得，鍵送信モジュールは perl で実装した．

4.2 モニタリング処理性能

上述の実験環境下において，どの程度の通信量までならパケットを取りこぼすことなく処理できるか定量的に評価した結果を表 2 に示す．実験方法は，クライアント間で UDP パケットを 1 分間に表 2 の設定値だけ送受信し，管理者端末で，実測値，処理パケット数，破棄パケット数をそれぞれ計測した．

表 2. 管理者端末での計測パケット数（/分）

出力先	パケット数		処理	破棄
	設定値	実測値	パケット数	パケット数
ターミナル	1200	1200	1180	0
	3000	2860	2776	84
	4000	3774	3518	226
	6000	-	4456	測定不能
ディスク	6000	5439	5439	0
	60000	29959	29939	20

表 1 から，出力先をターミナルとした場合，破棄するパケットも少なく正常にモニタリングできているのは実測値が 2860 パケット/分のときまでと判る．実装したモニタリングシステムを用いた即時解読の場合には，約 24.4kbps(=2860×512bit÷60sec) まで処理できることになる．また，本システムを使用せず tcpdump を使い直接モニタリングした結果を表 3 に示す（設定値 4000）．

表 3. 管理者端末での計測パケット数

データサイズ[bit]	実測値	処理パケット数	破棄パケット数
128	3939	3939	0
2048	3700	3170	530

表 3 で，同じパケット数にもかかわらずデータサイズの大きさによる破棄パケット数の相違と，表 2 のディスク書き込みでは破棄パケットはほぼ発生しないことから，破棄パケット数は tcpdump の性能に依存する部分が多い．

4.3 鍵の取得に関する評価

クライアント間の Re-Key を有効にし，4.2 節と同様の実験をおこなった．なお，鍵の送受信にかかる時間，すなわち，鍵取得モジュールが鍵を要求してから受信完了までに要する時間は約 13ms である．

表 4. 管理者端末での計測パケット数（Re-Key 有効）

出力先	パケット数		処理	破棄
	設定値	実測値	パケット数	パケット数
ターミナル	1200	1173	1173	0
	3000	2852	2852	0
	4000	3969	3777	192

表 2 と表 4 の比較により，Re-Key の発生有無が，解読可能なパケット数に影響を与えないことが判る．それは，設定値 4000 パケット/分の通信量からの破棄パケット数の増加など破棄パケット数の類似性から言える．

5. おわりに

本稿では IPsec による暗号化通信を対象として，鍵の自動取得機構を装備した暗号化通信をモニタリングするシステムを提案した．また，プロトタイプシステムを用いて，その処理性能に関する実験結果を示した．今後，以下の問題解決を検討していく予定である．

- 今回の実装では，トランスポートモードにおける 2 者間の通信のみを対象としたが，トランスポートモードの他者間およびトンネルモードへの対応の検討．
- 解読したデータの扱いや鍵管理など，本モニタリングシステムの運用方法．

6. 参考文献

[1] 松下電工株式会社，NetCocoon Analyzer, http://www.nais-netcocoon.com/jp/home/home_h.html

[2] 馬場達也，マスタリング IPsec，株式会社オライリー・ジャパン，2001