Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

セッション管理サーバによる SOME/IP セキュリティの検討

福田國統[1]　安永貴仁[1]　礒山芳一[1]

概要：車両システムの情報化にともない，車両内の複数の機能（サービス）を組み合わせて様々なサービスを実現する，サービス指向アーキテクチャが提案されている．その方法として，Scalable service-Oriented MiddlewarE over IP（SOME/IP）が標準化されている．しかし SOME/IP では，メッセージ認証が無いことや，通信の記録を一元管理できないため，攻撃のリスクがある．本論では，これらのセキュリティリスクを SOME/IP のセッションの開始と終了を管理するセッション管理サーバにより低減させる方法を提案する．共有鍵の共有方法やログの取得における，コストとリスクのバランスを考察し，提案手法が優れていることを示す．

キーワード：SOME/IP，鍵管理，デジタルフォレンジック，ゼロトラスト，セッション管理サーバ

A study on session management server for SOME/IP cybersecurity

KUNITO FUKUDA[1]　TAKAHITO YASUNAGA[1]
YOSHIKAZU ISOYAMA[1]

Abstract: With the informatization of vehicle systems, a service-oriented architecture that combines multiple functions (services) in a vehicle to achieve various services has been proposed. Scalable service-Oriented MiddlewarE over IP (SOME/IP） has been standardized as a way to achieve this. However, SOME/IP is vulnerable to at risk of attack due to a lack of message authentication and an inability to centrally manage communication records. In this paper, we propose a method to reduce these security risks by means of a session management server that manages the start and end of SOME/IP sessions. We consider the balance between cost and risk in log acquisition and shared key sharing and show the superiority of the proposed method.

Keywords: SOME/IP, Key management, Digital Forensics, Zerotrust, Session Management Server

1. はじめに

近年では，利用者が様々なアクセスポイントからネットワークにアクセスし，サービスを利用する機会が増えている．また，IoT 機器の普及が進み，ネットワークを構成するノードの分散化がより一層加速している．このようなネットワーク環境では，ネットワーク構造における内外の境界が不明確となり，境界上で動作するセキュリティ手法では防御が不十分となりやすい．

そこで，ゼロトラストの概念が注目されている．ゼロトラストは，信頼しないことを前提に動作するセキュリティの概念である．そのため，データやサービスを利用するたびに通信の検査を行い，そのログを保存する．

また，車両における通信は，エンジンやステアリング等の機能と強く紐づけられた ECU 同士が行い，それらの通信関係は製造時に静的に定義されてきた．しかし，車両における運転支援システムや自動運転等では，各種サービスで

1 住友電気工業株式会社
Sumitomo Electric Industries, Ltd. 1-1-3, Shimaya, Konohana-ku, Osaka, 554-0024, Japan

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

あるセンサー，エンジン，ブレーキ，ステアリングの機能を組み合わせて実現されるため，それぞれの機能を動的に変更，組み合わせることで，その保守や改良，利便性が高まる．つまり，それらの基盤となる通信プロトコルは，各目的で車両内のサービスを使用する高い利用性，サービスの再利用が求められる．そこで，BMW を中心として 2011 年に車両向けのサービス指向アーキテクチャとして SOME/IP が開発され，その後 AUTOSAR により標準化された [1] [2] [3]．SOME/IP は，車載ミドルウェアであり，データのシリアル化や，リモート呼び出し手順などを提供する．これにより，サービス単位での動的な運用が可能となる．

上記の利点を持つ SOME/IP ではあるが，一方でセキュリティ上の問題が指摘されている [4] [5] [8]．それは，SOME/IP のメッセージが暗号化や認証を行わないためである．

そこで，我々は SOME/IP においてゼロトラストの概念を導入し，セキュリティ機能を付与する方式（SMS-method）を提案する．SOME/IP においてゼロトラストを実現する方法として，セッション管理サーバ（SMS）を導入する．SMS は，共有鍵をセッション毎に交付し，セッションの開始と終了を管理する．また，そのときにログを残すことでデジタルフォレンジックの機能を有す．本論では，SOME/IP 上で動作する SMS の概要を示し，その機能が車載イーサネットにおいてセッションの管理とログ管理に適していることを示す．

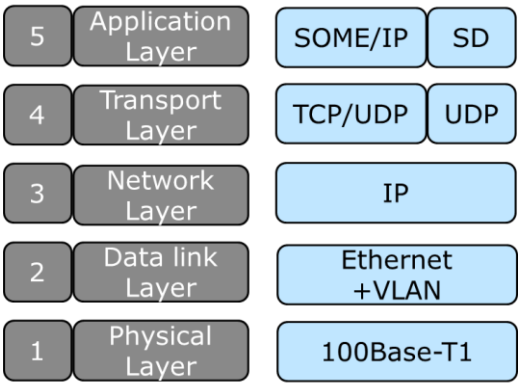

図 1　SOME/IP のプロトコルスタック

2. SOME/IP の仕様

この章では，提案する SMS の動作と関連する SOME/IP (-SD)の動作，および，課題について解説する．

2.1 概要

サービスを組み合わせてシステムを構築するには，求めるサービスが利用可能な形でネットワーク上に存在するか確認することが重要となる．SOME/IP では，サービス探索，状態の確認を行う Service Discovery (SD)の機能が用意されている [3].

SOME/IP のプロトコルスタックを図 1 に示す．TCP/IP プロトコルの土台の上に，SOME/IP 通信が行われる．また，SOME/IP は Linux などの複数の OS において利用が可能である．

SOME/IP を用いて通信する方法として，RPC(Remote Public Control)と Publish/Subscribe が用意されている．ここで，Publish/Subscribe にて使用される SOME/IP-SD の通信シーケンスを図 2 に示す．サービスを提供するサーバと，サービスを利用するクライアントは，図 2 に示すサービスディスカバリ処理を経て，セッションを確立する．図 2 の左側は，サービスを申し出るサーバからの OfferService メッセージからシーケンスが開始される．そのサービスを必要とするクライアントが OfferService メッセージの返信として SubscribeEventgroup メッセージを送信し，サービスの利用を申請する．SubscribeEventgroup メッセージを受信したサーバが，その返答として SubscribeEventgroupAck/Nack メッセージを返信し，その後，SOME/IP のセッションが開始される．図 2 の右側は，クライアントがサービスの探索として FindService メッセージを送信する場合である．求められるサービスを提供可能なサーバが FindService メッセージに対して OfferService メッセージを返信し，その後は左側同様の通信により，SOME/IP のセッションが開始される．RPC では，OfferService メッセージ後の SubscribeEventgroup メッセージ以降の SOME/IP-SD のメッセージが省略され，Request/Response などの SOME/IP 通信が開始される．

セッションを終了する場合は，専用のメッセージが SOME/IP-SD で用意されている．サーバがセッションの終了をクライアントへ通知するには，StopOfferService メッセージを送信する．一方，クライアントがサーバへ終了を通知する場合には，StopSubscribeEventgroup メッセージを送信する．

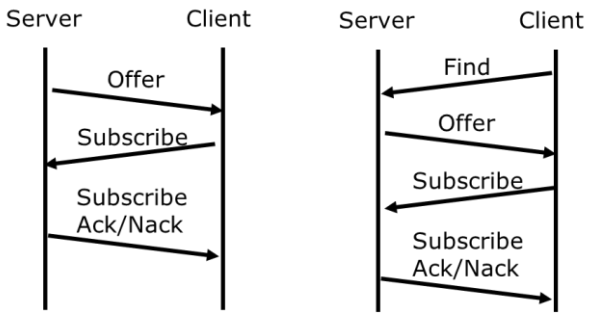

図 2 Publish/Subscribe における SOME/IP-SD のシーケンス

2.2 SOME/IP の課題

SOME/IP では上記の手続きにより，動的にサービス環境が変化する場合にも，必要とする機能を利用することが出来る．一方で，SOME/IP プロトコルでは，なりすまし攻撃が容易との指摘がなされている．SOME/IP では，メッセージ内の各種パラメータにより送信元のサービスを識別する．

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

しかし，SOME/IP のメッセージ自体は平文で送受信がなされため，盗聴による解析が可能である．また，識別はあくまでサービスの利用者と提供者を繋ぐ目的で存在しているため，送信相手が正規のサービスか否かは確認しない．つまり，SOME/IP の仕様上は，不正規のサービスが，正規のサービスになりすました場合，パラメータが適切であれば，受信側はそれを受信し，処理してしまう．その場合，改ざん [5]，DoS 攻撃，サービスの偽造，セッションの乗っ取り [8]などの危険性が指摘されている．これを防ぐ一つの方法として，各 SOME/IP に対するメッセージ認証がある．

メッセージ認証の方法として暗号化が考えられる．しかし，従来の共有鍵方式では，共有鍵が漏洩した場合，次の鍵更新までセキュリティとして機能せず，また即時に鍵を更新するのも難しいと思われる．また，公開鍵暗号を使用する場合には，各 ECU に負荷の大きな公開鍵暗号を処理する能力が求められ，コスト高となる課題がある．

さらに，多数のサーバやクライアントが分散して配置される場合，Ethernet ネットワークでは，すべての通信を傍受する機器を用意するのは難しく，サービスの利用や停止のすべての記録の一元管理や，不正なアクセスやサービスの利用の記録が難しいと思われる．また，複数のロギング装置を用意した場合に，攻撃のシーケンスを再現するためには，全ロギング装置の時刻同期が必須となり，コスト高となる．

2.3 先行研究

Marco らは，SOME/IP におけるメッセージ認証によるセキュリティを検討している [4]．そのとき，合わせて，SOME/IP に IPSec や TLS を導入する場合の効果も調査した．各サーバとクライアント間の鍵管理を提案し，SOME/IP の実装例である vsomeip [7]への適用性を確認している．しかし，送受信間で認証を行う機構であり，通信にともなうメッセージのログを残す機能が考慮されていない．

一方で，IoT とクラウドなどのゼロトラスト環境における鍵管理方式が検討されている．例えば鈴木らは，クライアントとサービス提供サーバ，そして鍵発行サーバで構成された三極構成の共有鍵管理方式を提案した [6]．この方式は，送受信間で鍵シードを交換することなく，第三者の仲介により安全かつ比較的軽量で鍵共有する方式と言える．また，セッションの開始に関するログを保存することが出来る．しかし，セッションの開始が管理されるのに対して，終了は管理されない．SOME/IP では，StopOfferService メッセージや StopSubscribeEventgroup メッセージをなりすまし，センサー情報などの車両動作に必要なサービスを停止させる攻撃が考えられるため，サービスの終了も開始同様，管理する必要があると考えられる．また，サービス提供サーバがクライアントを正規か判定するが，SOME/IP ではサーバは新規のクライアントを判定することが困難である．

そこで，第三極の構成を基本としながら SOME/IP の仕様を考慮し，SMS の動作を検討した．上記課題を解決する SMS を用いた SOME/IP の SMS-Method を以下に示す．

3. SMS の動作

3.1 SMS の概要

前述のセキュリティ上の問題を解決するために，今回我々は SMS を SOME/IP(-SD)に導入する手法を提案する．サーバとクライアント間で SOME/IP 通信を行う前に，それぞれが SMS と通信し，SMS が信頼できる接続先か判断し，必要な情報をサーバとクライアントに提供する．概要を図 3 に示した．

概要としては，以下の 2 ステップでサーバとクライアント間の安全な通信を確保する．まず，サーバとクライアントは，SMS と通信を行い，それぞれが正当な ECU とサービスか判定される．正当だと判断された場合，SMS が，サーバとクライアント間の通信で使用する共通鍵を配布する．この共通鍵は，各 ECU が持つ固有 ID により暗号化されて配付される．固有 ID は，各 ECU 本体と SMS 上の接続先リストに記録されている．この鍵の発行は，セッション毎に行う．この鍵の発行までのシーケンスは，図 2 の OfferService メッセージ，あるいは FindService メッセージから OfferService メッセージの部分に対応する．RPC と Publish/Subscribe 両方に対応するために，OfferService メッセージの後の通信は，サーバとクライアント間で配布した共通鍵を用いて暗号通信を行う．

正当な ECU の認証は表 1 に示す各機器が内部に保持する情報を用いて行われる．サーバとクライアントは，各 ECU の固有 ID，ECU の識別子，SMS の IP アドレスとポート番号で構成されるエンドポイント情報を保持している．SMS は接続先リストを保持しており，サービス毎にサーバとクライアントのリストが記録されている．接続先リストの内容を表 2 に示す．接続先リストは，サービス ID，エンドポイント情報，ECU 識別子，固有 ID，そのサービスに対する機能（サーバかクライアントか）の情報で構成されている．エンドポイント情報のみ，各 ECU から送信されるメッセージによって更新され，それ以外の情報は，静的に保持する情報である．

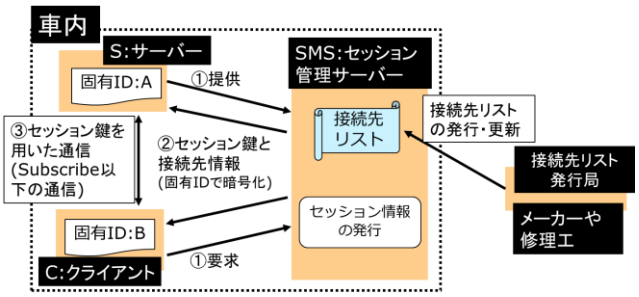

図 3 SMS の概要

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

表 1 各機器が内部に保持する情報

機器	保持する情報	概要
サーバ＆クライアント	サーバの固有ID	各ノード固有のID。機器認証に使用
	ECU識別子	各ECU固有の識別子
	管理サーバのエンドポイント	管理サーバのエンドポイントは、静的に設定
管理サーバ	接続先リスト	管理サーバが接続先を認証する際に使用

表 2 接続先リスト

サービスID	エンドポイント情報	ECU識別子	固有ID	Server or Client
0x0001	～～～～～～～	0x038ef839a	iaiehdij29	S
	～～～～～～～	0x038ef840a	93ogealfih	S
	～～～～～～～	0x038ef841a	ie9w3j27aej	C
	～～～～～～～	0x038ef842a	keio92o94lu	C

3.2 接続先リストの更新

接続先リストは, 基本的には, 車両に搭載されている ECU に基づき工場出荷時に登録されるが，車両利用期間の経過により, ECU の交換や新しい機能を持つ ECU やサービスが追加される場合には，更新されなければならない．更新に際しては，診断ツール等から直接更新する方法や，車両が外部ネットワークと接続が可能な場合，SMS がネットワーク経由で管理元に問い合わせ，新しいリストをダウンロードする手法が考えられる．

3.3 セッション確立の手順

この節では，SMS を介したサーバとクライアントのセッション確立の手順を説明する．図 4 にセッション確立のプロセスを示した．図 4 の上段がサーバからクライアントへサービスの提供を申し出る，Offer タイプの通信．下段はクライアントからサービスを探索する Find タイプの通信を示している．

Offer タイプの通信手順は以下のようになっている．

1. サーバから SMS へサービスの提供をユニキャストで申請する．サービスの提供は，サービス ID，サーバの ECU 識別子，サーバのエンドポイント情報(EIs)が含まれている．このメッセージには，サーバの固有 ID(IDs)で作製されたメッセージ認証コード(MAC）が付与されている．
2. SMS は，そのサービスが適切か判断する．不適切と判断した場合は，メッセージを破棄する．判定は，リストを用いた判定と MAC 認証による判定が行われる．リストの判定は，サービス ID，固有 ID の情報が参照され，接続先リストに登録されているか確認する．その後，SMS からクライアントへサービスを利用するか確認の為，マルチキャストでメッセージを送信する．このメッセージには，MAC を付与しない．
3. そのサービスを必要とするクライアントが SMS に返信する．クライアントから返信があった場合，SMS がセッション番号(SN)とセッション鍵(SK)を作成する．セッション番号とセッション鍵は，ランダムに作製される．
4. SMS がクライアントにセッション番号と，(EIc||SK)⊕H(IDs||SN)のメッセージを送信する．H(x)は，ハッシュ関数である．つまり，クライアントのエンドポイント情報(EIc)とセッション鍵は，サーバの固有 ID とセッション番号を入力としたハッシュ値と，排他的論理和を取った形で送信される．これにより，サーバの固有 ID を知らない第三者は，メッセージの偽造や解読が困難となる．OfferService メッセージが否定された場合は，セッション鍵に否認を示す値を代入することで，サーバに通知する．
5. SMS がクライアントに, 4 と同様のメッセージを送信し，サーバのエンドポイント情報とセッション鍵を通知する．ただし, 固有 ID はクライアントのものが使用されている．
6. サーバはステップ 4 で受信したメッセージを検証する．具体的には，セッション番号とサーバの固有 ID から，ハッシュ値を作成し，メッセージを復元する．セッション度にランダムに生成されるセッション番号を用いることで，リプレイ攻撃を防止している．続いて，復号したメッセージから，セッション鍵を得る．クライアントでも同様に，クライアントの固有 ID を使用してメッセージを復元し，セッション鍵を得る．その後，サーバとクライアントが，セッション鍵を用いてメッセージを暗号化し，通信する．

Find タイプの通信手順は以下のようになっている．

1. クライアントが SMS へサービスを要求する．サービスの提供は，サービス ID，サーバの ECU 識別子，サーバのエンドポイント情報が含まれている．このメッセージは，クライアントの固有 ID を用いた認証情報である MAC が付与されている．
2. SMS は，クライアントがそのサービスを要求するのが適切か判断する．不適切と判断した場合には，破棄する．リストの判定は, サービス ID, 固有 ID の情報が参照され，接続先リストに登録されているか確認する．その後，SMS からサーバへサービスの要求をマルチキャストで送信する．このメッセージには，認証情報である MAC を付与しない．
3. そのサービスを提供可能なサーバが SMS に返信する．サーバから返信があった場合，SMS がセッション番号とセッション鍵を作成する．

その後のステップ 4~6 は，Offer タイプの通信手順と同様に処理を行う．

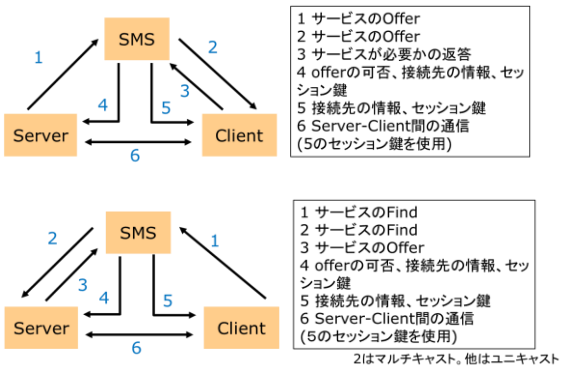

図 4 セッション確立の手順

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

3.4 セッションの終了

ここで，セッションの終了をサービスインスタンスの終了と定義する．

サーバ側のサービスインスタンスが終了する場合の処理手順を図 5 に示す．まず，サーバが SMS へ StopOfferService メッセージを送信する．StopOfferService メッセージは，セッション鍵を用いて暗号化されている．サーバは，StopOfferService メッセージを送信後にクライアントのエンドポイント情報やセッション鍵を破棄する．続いて，StopOfferService メッセージを受信した SMS は，セッション中のすべてのクライアントへ StopOfferService メッセージを送信する．StopOfferService メッセージを受信したクライアントは，エンドポイント情報と共通鍵を破棄する．以上で，セッションが終了される．

クライアント側のサービスインスタンスが終了する場合の処理手順を図 6 に示す．まず，クライアントが SMS へ StopSubscribeEventgroup メッセージを送信する．StopSubscribeEventgroup メッセージは，セッションで使用したセッション鍵で暗号化されている．クライアントは StopSubscribeEventgroup メッセージを送信後，サーバのエンドポイント情報やセッション鍵を破棄する．StopSubscribeEventgroup メッセージを受信した SMS は，セッション中のサーバへ StopSubscribeEventgroup メッセージを送信する．StopSubscribeEventgroup メッセージを受信したサーバは，そのクライアントのエンドポイント情報と共通鍵のみ破棄する．

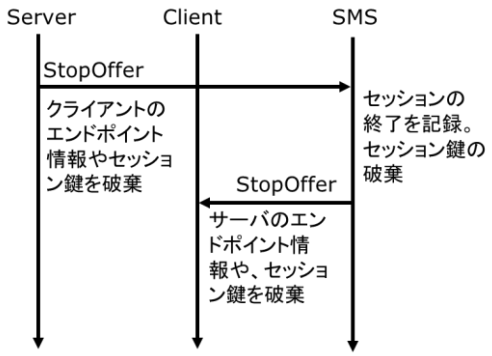

図 5 StopOfferService によるセッションの終了

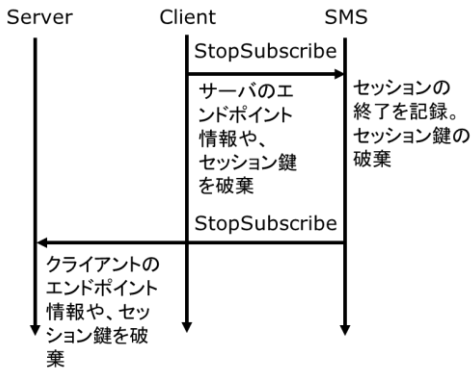

図 6 StopSubscribeEventgroup によるセッションの終了

3.5 SMS によるログ保存

SMS は，大きく分けて 2 種類のログを記録する．1 つは，セッションの確立と終了のログである．2 つめは，SMS が受信した確立・終了に失敗したメッセージのログである．

1 つめのセッションの確立と終了では，サービス ID，セッションに参加したノードの ECU 識別子，セッション確立時間，セッション終了時間を記録する．表 3 に記録するログ形式を示す．セッション確立時間は，SMS にセッション鍵を配布したタイミングとなる．セッション終了時間は，StopOfferService メッセージか StopSubscribeEventgroup メッセージを受信した時間を記録する．ECU の識別子から，異常なメッセージを送信元した ECU の特定を行う．既存の ECU が，他の ECU が送信する不正規なメッセージを送信したか判定する材料となる．各時間の情報から，攻撃メッセージが発生した時刻を知ることができ，さらに，攻撃 ECU がどの段階で異常な動作を始めたのか，その原因分析に利用できる．

2 つめは，SMS が破棄したメッセージの受信時間と送信元 ECU の記録である．表 4 に記録するログ形式を示す．これは，攻撃開始時刻を調査するために重要な情報となる．セッションの確立や終了を伴わない攻撃を記録可能となる．つまり，影響が表面化する前の攻撃の情報収集（開始時刻や機器）であり，攻撃成功までの手法の解析が可能となる．攻撃開始から，攻撃が成功するまでの時差がある場合に，有効な記録となる．

これらのログを SMS が記録することで，正常状態への復帰方法や，類似した攻撃への防御機構を検討する際の情報を提供できる．

表 3 セッションの確立と終了のログ形式

サービスID	ECUの識別子	開始時刻	終了時刻
Serv1	ECU1	ts1	te2
	ECU2	ts1	te1
	ECU3	ts2	te2

表 4 確立・終了に失敗したメッセージのログ形式

メッセージ(バイト列)	送信元の識別子	受信時刻
M1	ECU1	t1
M2	ECU2	t2

3.6 マルチキャストへの対応

ここまでの議論では，サーバとクライアントが 1 対 1 で通信を行う場合を考えてきた．しかし，SOME/IP では，あるサーバの機能が複数のクライアントへ提供される場合がある．実際，SOME/IP-SD の OfferService メッセージは，あるクライアントとセッションを確立したとしても，定期的に送信される．つまり，常に新規のクライアントの接続を受け付けている．そこで，SMS が導入された場合のマルチ

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

キャスト時の動作を検討した．概要としては，マルチキャストの場合は，マルチキャストに参加する機器同士で同一のマルチキャストセッション鍵を使用する．

サーバとクライアント間がユニキャストで通信している状況において，正規の機器が乗っ取られた場合を考える．ユニキャスト通信のため，乗っ取った機器として振舞うことは可能だが，通信相手になりすましてメッセージを送信することは，他の受信者が存在しないため困難である（図 7 上部）．

一方，マルチキャストに参加する機器が乗っ取られた場合，攻撃範囲が拡張される．その状態を図 7 の下部に示す．マルチキャストでは，セッション内に複数の機器，サーバ(S)とクライアント C1，C2 が参加している．このマルチキャストにおいて，C1 が乗っ取られ，S になりすます場合を考える．C1 から送信された不正規のメッセージは，マルチキャスト用のセッション鍵（KM1）で暗号化されているため，正規の機器の C2 は不正な信号と判断できず，サーバからのメッセージとして処理してしまう．一方，本来，そのメッセージを送信するなりすまされたサーバ S は，そのメッセージを受信すると，異常と判断することが出来る．このとき，異常を検知したサーバは，現在自分がなりすまされていること，つまりマルチキャストに不正機器が参加していることを，マルチキャストの参加者に通知する．これにより参加者は，マルチキャストにおける異常を把握することが出来る．通知が行われた場合，サーバとクライアントは，1 対 1 のユニキャスト通信に切り換える．このユニキャストに用いられるセッション鍵（KUx）は，サーバとクライアント間の組合せで個別の物を使用するため，なりすましのメッセージを受信処理することを防げる．

異常を検知したサーバは，車両の利用者にも異常を通知し，車両の点検を促す．ユニキャストに切り換えることにより，クライアントへ個別にメッセージを送信しなければならないため，サーバの送信負荷が上昇すると考えられる．そのため，この方式での切り換えた後は，メッセージの送信頻度が低下することも考えられる．しかし，利用者が車両を安全に停車できる状況までの緊急処置的な通信モードとして想定おり，停車までの最低限の通信を安全に保つことを目的としている．

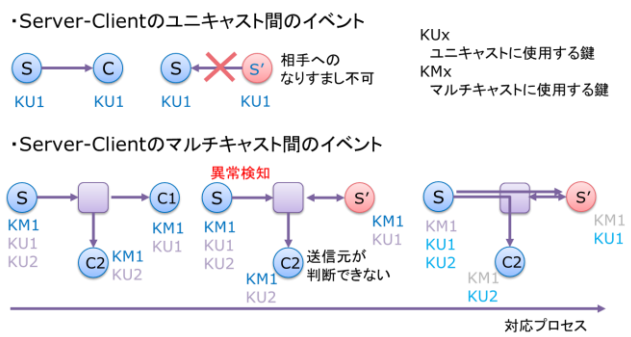

図 7 マルチキャストへの対応

3.7 SMS の構成

SMS は，セッション毎のセッション鍵の発行や，ログを記録するに十分な能力が必要となる．また，ネットワーク構造において通信が集中することが望ましい．そのため，処理能力が高いセントラルゲートウェイ等に搭載するのが有効と考えられる．

4. SMS の利点

この章では，SMS を導入した場合の利点を記述する．SMS の利点は大きく分けて，秘密漏洩に対するリスク低減と，SMS によるデジタルフォレンジックがある．これらについて説明をする．

4.1 秘密漏洩に対するリスク低減

SOME/IP(-SD)では，メッセージ認証の機構がないため，不正機器からの攻撃メッセージの送信や，なりすましが行われるリスクがある．

この対策として，各メッセージの暗号化が考えられる．この際に，公開鍵暗号と共有鍵暗号が考えられる．公開鍵を用いた場合，各 ECU に公開鍵暗号の機能が必要となる．また，その暗号化処理が必要となる．これらの処理を全メッセージに適用するには，ECU への負荷が大きく，また，十分なリアルタイム性を保障することが困難となる．

共有鍵暗号では，暗号化処理は公開鍵と比較して軽量という利点がある．一方で，鍵の管理方法が問題となる．共有鍵が漏洩した場合，その共有鍵を利用する機器になりすまされる危険性がある．各通信関係別に異なる共有鍵を使用する場合，各 ECU は、通信関係の数だけ鍵を保持し，使い分ける必要があり，鍵の管理が煩雑化する．

提案する SMS-method では，SOME/IP-SD のメッセージ認証に各 ECU の固有 ID に基づく認証情報である MAC を付与し，DoS やなりすましを防御している．これにより，SOME/IP のサービス指向プロトコルは，動的に共有鍵であるセッション鍵を共有することが可能である．共有鍵方式のため，暗号化による負担が比較的小さい．また，セッション毎に新しいセッション鍵を交付するため，解読を困難としている．もし，セッション鍵が漏洩した場合でも，セッションの再確立とともにセッション鍵が更新されるため，次のセッションではなりすましが困難となる（図 8 中央）．

図 8 右端に示すように固有 ID が漏洩した場合は，なりすましの防御が困難となる．しかしそれでも，セッションの確立は，漏洩した固有 ID の ECU 以外では困難となる．つまり固有 ID が漏洩した場合にも，影響の範囲が限定的となる．

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

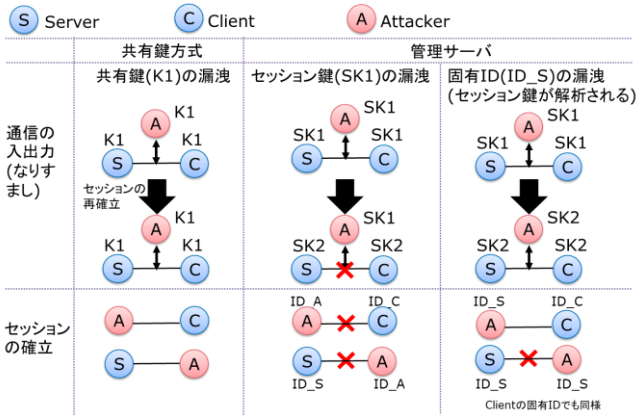

図 8 秘密漏洩に対するリスク低減

表 5 各鍵管理方法のコストとリスクの比較

	公開鍵	共有鍵	管理サーバ
既存のサーバ・クライアントへの影響	×全ECUに公開鍵暗号処理実装	△全ECUに共有鍵暗号処理実装	△全ECUに共有鍵処理実装
鍵管理の負担	○公開鍵のみ	△サービス数と同じ個数の鍵を管理	○固有IDのみ
鍵漏洩時のリスク	×全サービスを攻撃可能	△漏洩したサービスのみ攻撃可能	○漏洩したECUまたは、該当のセッションのみ攻撃可能

4.2 SMS によるログの記録

通常の SOME/IP(-SD)では，ロギング機能は規定されていない．しかし，攻撃元 ECU の特定や除去，乗っ取られた ECU の正常状態への復帰，受けた攻撃の分析，防御手法の構築のためには，攻撃メッセージの内容，攻撃が行われた時刻，攻撃元の機器の情報等が必要であり，SOME/IP(-SD)には，ログの保存機構がある事が望ましい．ここで，ログの記録方法として，3 つの手法が考えられる．

1 つめは，各 ECU が自身の送受信を記録する手法である（図 9 左）．メッセージや接続先の認証を各 ECU が個別に行う場合には，各 ECU でセッションの開始と終了が管理されるため，この方式でのログ方式を使用することが適切と考えられる．この手法では，全 ECU にログ機能を付与する必要があるため，導入に高コストと思われる．また，分散しているためログ機能の保守が煩雑となる．また，それぞれの ECU が同期した時刻を持たなければ，各 ECU の記録を比較し，時系列的な変化をログデータから解析することが困難である．

2 つめは，ログ専用のロガーECU を配置し，各 ECU のメッセージを記録する手法である（図 9 中央）．各 ECU にログ機能を実装する必要がないため，導入に際して変更点が少ない利点がある．一方，SOME/IP(-SD)では，メッセージはブロードキャスト送信でないため，サーバとクライアン間でのみ通信されるメッセージをロガーに集約する必要がある．つまり，記録用にロガーにメッセージを転送しなければ，正確なログを記録することが出来ない．そのため，車内ネットワークのトラフィック量が 2 倍に増加し，通信帯域を圧迫，正常な通信を妨害する危険性がある．また，不正機器がロガーへメッセージを転送しない場合には，攻撃メッセージが記録されないという課題が残る．

3 つめは，提案している SMS-method である（図 9 右）．すべての通信がロガーの機能を持つ SMS を経由するため，トラフィック量の増加は抑えられる．SOME/IP-SD では固有 ID を用いた MAC 認証，SOME/IP 中では，セッション鍵を用いた暗号化がなされているため，不正機器がロガーを経由せずに各 ECU と通信することは困難である．本方式では，SMS への攻撃が予測されるが，その際は SMS がすべての攻撃を記録する．既に述べたように，SMS は，セッションが成功した場合に限らず，不正と判断したメッセージも記録するため，固有 ID を総当たりで割り出す手法が取られた場合には，その初期の段階で記録を残すことが出来る．

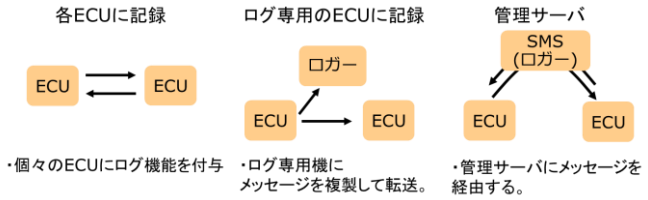

図 9 記録方式の比較

表 6 ロギング手法の比較

	各ECUで記録	専用ロガー	管理サーバ
既存のサーバ・クライアントへの影響	×：全ECUのログ記録、時刻同期機能の追加	△：専用ロガーへメッセージを転送	△：シーケンスの追加
ログによる通信量の負荷	○：変化なし	×：二倍に増加	△：セッション開始と終了時のメッセージが増加
記録	○：全ログを記録	×：専用ロガー経由のメッセージのみ記録	○：セッション開始と終了に関する全ての記録が可能
追加機器	○：なし	△：専用ロガー	△：管理サーバ

5. まとめ

SOME/IP におけるセキュリティ対策を検討し，セッションの管理機構を提案した．セッションの管理には，以下の利点があることを示した． 1 つ目は，秘密情報である，セッション鍵や ECU の固有 ID が漏洩した場合のリスクの低減である．セッション毎の共有鍵の更新により，漏洩後の被害を限定できる．また，各 ECU が保持する固有 ID が漏洩した場合も，他の ECU になりすましてセッションを確立することが出来ない．

Vol.2021-ARC-244 No.25
Vol.2021-SLDM-194 No.25
Vol.2021-EMB-56 No.25
2021/3/25

2 つ目は, デジタルフォレンジックの機能である．各 ECU にログ機能を搭載する場合や，専用ロガーを設置する場合に比べて，各機器の負担を少なくセッション時の通信先を記録できる．

このセッション管理サーバの導入により，セキュアな SOME/IP を車両へ導入可能と考えられる．今後は，実装により，想定通りにセッション管理サーバが動作し，攻撃を防御可能か検証する．また，導入による各機器の負荷や，トラフィックへの影響を調査する．

参考文献

[1] L. Völker, “Scalable service-Oriented MiddlewarE over IP (SOME/IP),” http://some-ip.com/index.shtml.

[2] Autosar, “SOME/IP Protocol Specification,” Release R20-11, Document ID 696: November 30, 2020.

[3] Autosar, “SOME/IP Service Discovery Protocol Specification,” Release R20-11, Document ID 802: November 30, 2020.

[4] M. Iorio, A. Buttiglieri, M. Reineri, F. Risso, R. Sisto and F. Valenza, “Protecting In-Vehicle Services: Security-Enabled,” IEEE VEHICULAR TECHNOLOGY MAGAZINE, 2020.

[5] Y. Keigo, H. Yoshihiro, A. Naoki, A. Shinichi, I. Fumiya, K. Shogo, U. Hiroshi, and H. Yoichi, “Study of intrusion detection method for SOME/IP,” SCIS2020, 2020.

[6] S. Nobuharu, S. Koji, Y. Takahiro, Y. Kenya, T. Toshio, Y. Masao, G. Masahito, and T. Shigeo, “Key Holding System Without Exchanging Common Key between Users,” 研究報告セキュリティ心理学とトラスト (SPT) 2020, 2020.

[7] AG, BMW, “GENIVI/vsomeip,” https://github.com/GENIVI/vsomeip.

[8] T. Kishikawa, T. Adachi, R. Hirano, Y. Ujiie, T. Haga, and H. Matsushima, “Flow-based IDS for Automotive Ethernet:,” SCIS 2021, 2021.