Computer Security Symposium 2017
23 - 25 October 2017

送信 ECU 上の不正送信阻止機構を回避する同期送信を使ったなりすまし攻撃とその対策

家平 和輝[1]　井上 博之[2,3]　石田 賢治[2]

概要：車載ネットワークで広く使用されている CAN（Controller Area Network）は，共有バス型であり，送信元アドレスを持たず，認証や暗号化の仕組みもないため，なりすまし攻撃に対して脆弱である．特定の送信先 CAN ID を送信する ECU（Electronic Control Unit）は網内に 1 台であることに着目して，なりすまし攻撃を正規の送信 ECU で検知し阻止する不正送信阻止方式が提案されている．本論文では，まず，なりすましメッセージを正規メッセージと同時に送信することで，この不正送信阻止方式でも検知できないなりすまし攻撃を提案した．次に，模擬環境での評価実験を行い，実現可能性を確認した．さらに，同時送信されたときの挙動を十分に考慮することで，提案した攻撃方式も検知し阻止できる不正送信阻止方式を提案し，同様に有効性を確認した．

キーワード：不正送信阻止，なりすまし攻撃，CAN，車載ネットワーク，ECU

Proposal and Countermeasures for Spoofing Attack to Prevent Unauthorized Message Transmission

KAZUKI IEHIRA[1]　HIROYUKI INOUE[2,3]　KENJI ISHIDA[2]

Abstract: The controller area network (CAN) widely used in in-vehicle networks employs a shared bus-type topology, lacks a source address, and lacks an authentication and encryption mechanism. Therefore, it is vulnerable to spoofing attacks. The number of electronic control unit (ECU) with certain destination CAN ID is only one in the network. With this constraint, we propose a method for preventing unauthorized message transmission via an authorized transmission ECU. To achieve this, we propose a spoofing attack that cannot be detected (even with this prevention method). This attack simultaneously transmits a spoof message and regular message. We conducted an evaluation experiment in a simulated environment and confirmed feasibility. Furthermore, by fully considering system behavior when transmitting simultaneous messages, we propose a method that can detect and prevent the proposed attack, and confirm the effectiveness of that method.

Keywords: preventing unauthorized message transmission, spoofing attack, CAN, in-vehicle network, ECU

1. はじめに

近年，制御システムの電子化に伴い，自動車には多数の ECU(Electronic Control Unit)と呼ばれる電子制御ユニットが搭載されている． ECU は他の ECU と協調するため，車載ネットワークの通信規格として広く普及している CAN(Controller Area Network)に接続されている．

CAN プロトコルはバス型のネットワークであり，送信元アドレスを持たず，暗号化や認証の仕組みがないことから，なりすまし攻撃に脆弱であり，問題視されている．CAN の攻撃事例として，CAN バスや OBD-II 診断ポートに対してなりすましメッセージを注入することで，メーターの表示改ざんや，ブレーキ等の不正制御が可能であることが報告[1][2]されている．

なりすまし攻撃の対策として，特定の送信先 CAN ID をもつメッセージを送信する ECU が一つであることに着目して，送信 ECU でなりすまし攻撃を検知し阻止する不正送信阻止方式[3][4]が提案されている．この方式を用いることにより，確実になりすまし攻撃を阻止することができる．この方式に阻止されないなりすまし攻撃はいくつか提案されている．しかし，提案されている攻撃方式は全て電気的改ざんを用いており，実現するには特別な CAN コントローラが必要であるため，脅威としては小さなものであった．本論文では，まず，なりすましメッセージを正規メッセージと同時に送信することで，通常の CAN コントローラで行える既存の不正送信阻止方式で検知できないなりすまし攻撃を提案する．次に，模擬環境での評価実験を行い，実現可能性を確認した．さらに，同時送信されたときの挙動を十分に考慮することで，提案した攻撃方式も検知し阻止できる不正送信阻止方式を提案し，同様に有効性を確認し

1 広島市立大学情報科学部
Faculty of Information Sciences, Hiroshima City University
2 広島市立大学大学院情報科学研究科
Graduate School of Information Sciences, Hiroshima City University
3 重要生活機器連携セキュリティ協議会 研究開発センター
Connected Consumer Device Security Council (CCDS)

た．

2. 関連研究

2.1 CAN プロトコルの特徴

CAN プロトコル[5]では，CAN バスを構成する CANL および CANH と呼ばれる 2 本の信号線を用いた差動通信を行っている．論理“1”を表す状態をリセッシブと呼び，論理“0”に対応する状態をドミナントと呼ぶ．もし，リセッシブとドミナントが同タイミングで送信されると，ドミナントが優先される．また，CAN では同じ論理のビットが連続して送信されることにより同期が取れなくなることを防ぐためにビットスタッフィングルールを適用している．そのため，同じ論理のビットが 5bit 連続するときは論理を反転したビットを 1bit 挿入する．

通常のデータを送信するメッセージをデータフレームと呼び，データフレームは図 1 に示すように CAN ID と DLC(Data Length Code)とデータ部と CRC から構成される．また，CAN ID と DLC の間に RTR と IDE と r0 というドミナント固定のビットが構成される．エラーを周知するメッセージをエラーフレームと呼び， 6bit のエラーフラグと 8bit のエラーデリミタ（リセッシブ）から構成される。

エラー制御方法として，ECU がエラーを検知するとエラーフレームを送信することで，他の ECU に知らせる．CAN コントローラではエラーを検出することで，送信エラーカウント値（TEC），受信エラーカウント値（REC）の値を増加させる．特に，送信 ECU がエラーを検出した場合は TEC が 8 増加され，受信 ECU がエラーを検出したときは REC が 1 増加される．CAN コントローラには図 2 に示す 3 つの状態があり， TEC，REC の値に応じて状態が遷移し，バスへのアクセスに制限がかかる．さらに，エラーフラグのレベルも変化する．表 1 にアクセスの制限と状態に応じて送信されるエラーフラグを示す．エラーアクティブ状態は通常状態と定められており，制限はかからない．エラーパッシブ状態はエラーを起こしやすい状態と定められていて，他の ECU の通信を妨げないために連続送信時の送信待機

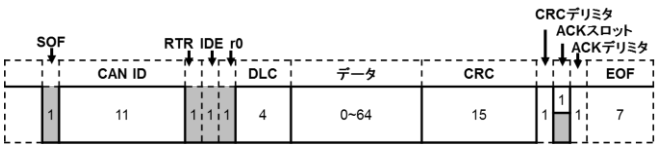


図 1 データフレームのフォーマット

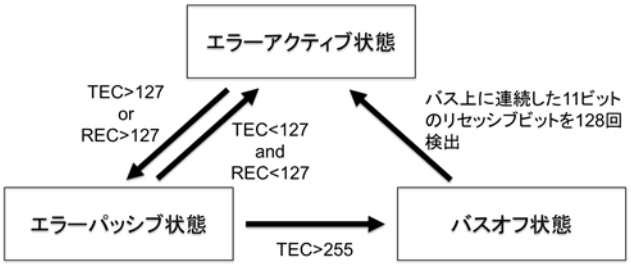


図 2 ECU の状態と遷移条件

表 1 ECU における状態に応じた制限

状態	CAN バスへの アクセス制限	エラーフラグ
エラーアクティブ	なし	6bit のドミナント
エラーパッシブ	連続送信時に 8bit 期間の送信待機	6bit のリセッシブ
バスオフ	送受信の禁止	送信不可

時間が通常より長く設定されている．さらに，エラー検出時もリセッシブのみを送信するため，他の ECU による送信を妨げることはできない．バスオフ状態はバス上の通信に参加できない状態と定められている．そのため，送受信すべてを禁止されている．

2.2 なりすまし攻撃に対する防御方式

なりすまし攻撃に対する防御方式として，CAN メッセージの周期性に着目した方式[6]や暗号化や認証の仕組み[7][8]が提案されている．しかし，正規の送信 ECU を無効化し，周期をなりすましたメッセージを注入する攻撃[9]が報告されている．そもそも，CAN ではペイロードを最大 8byte と容量が少なく，通信速度も最大 1Mbps と遅いため暗号化や認証の仕組みを導入するには十分でなく，これらの方式は実用的ではないと言える．

2.3 送信 ECU による不正送信阻止方式

CAN における不正送信阻止方式として，特定の送信先アドレスをもつメッセージを送信する ECU が一つであることに着目して，送信 ECU でなりすまし攻撃を検知し阻止する方式[3][4]が提案されている．この方式の基本的なアイデアとして，各 ECU がバス上に流れるデータを監視し，自身が送信するメッセージの ID をもつメッセージ，すなわちなりすましメッセージが他から送信されるのを阻止する．

具体的な方法として，文献[4]で提案されている方式では，正規の送信 ECU にて送信中を除いて受信したすべてのメッセージに対して，r0 ビットを受信したタイミングで受信 ID が当該 ECU の送信 ID リストに含まれていないかを確認する．ここで，受信 ID が送信 ID リストに含まれていた場合は不正送信と判断し，次ビットの DLC を受信したタイミングでエラーフレームの送信を開始する．これにより，なりすましメッセージがエラーとなり，送信が阻止される．この方式では，なりすましメッセージが CAN の仕様に準拠した CAN コントローラから送信される場合にのみ成立し，仕様に準拠していない場合は検知できない可能性がある．

2.4 送信 ECU による不正送信阻止を回避する攻撃方式

文献[3][4]で提案されている不正送信阻止を回避する攻撃方式がいくつか提案されている．送信 ECU と受信 ECU のビットを受信するタイミング（サンプルポイント）の違いに着目し，受信 ECU のサンプルポイントに合わせて CAN バスを構成する 2 線間の電位差を改ざんすることで，検知されないなりすまし攻撃[10]が提案されている．この方式

では，正規の送信 ECU が送信したメッセージを改ざんするため，不正送信阻止を回避することができる．電位差を改ざんすることにより，エラーフレームで上書きできないフレームを用いた不正送信阻止を回避する攻撃方式[11]が提案されている.この方式で送信されたなりすましメッセージはエラーフレームで上書きすることができないため，送信を阻止することはできない．ただし，検知することは可能である．文献[4]で提案されている実装方法では，ECU の状態に応じてエラーフラグが変わる仕様となっていた．そこで，電気的改ざんを用いて正規の送信 ECU の送信したメッセージをエラーにすることで，エラーパッシブ状態に遷移させ，なりすましメッセージの送信を無効化できないようにする攻撃[12]が報告されている．ただし，この方法は実装方法に依存しているため，常にアクティブエラーフレームを送信する実装であれば，この攻撃は有効ではない．

これらの攻撃方式全てにおいて，不正送信阻止を回避することができるが，CAN の仕様に準拠した CAN コントローラだけでは実現できない．

2.5 エラーカウント値を増加させるバスオフ攻撃

正規の送信 ECU のエラーカウント値を増加させる攻撃として，正規のメッセージと以下の条件を満たす攻撃メッセージを同時に送信することでビットエラーを誘発させる，バスオフ状態に遷移させるバスオフ攻撃[12]が提案されている．

条件1. 正規の送信 ECU と同一の CAN ID であること．

条件2. 正規の送信 ECU の送信したリセッシブビットに対して，攻撃者 ECU がドミナントビットを送信する．ただし，これ以前のビットは全て同一のものとする．

図 3 に示すように，正規の送信 ECU と攻撃者 ECU が同時に送信を開始する．その後，送信ビットが異なるタイミングで正規の送信 ECU がビットエラーを検出し，TEC が 8 増加される．この攻撃を繰り返し行うことにより，TEC>255 にして，正規の送信 ECU をバスオフ状態に遷移させる．この攻撃では攻撃者 ECU と正規の送信 ECU がエラーアクティブ状態であるときは再送も同時に行われるため，条件 (A) を満たすとき，一度の攻撃で正規の送信 ECU をエラーパッシブ状態に遷移させることができる．

$$\lfloor 攻撃者ECUのTEC/8 \rfloor \leq \lfloor 正規の送信ECUのTEC/8 \rfloor \quad (A)$$

バスオフ攻撃には，正規の送信 ECU をエラーパッシブ状態に遷移させると同時に攻撃に利用したメッセージが送信される特徴がある．攻撃に利用したメッセージが送信されているときの波形図を図 4 に示す．バスオフ攻撃では，正規メッセージと攻撃メッセージを同時に送信する必要がある．そこで，同時にメッセージを送信する方法として，正規メッセージの送信直前に内容に意味を持たないメッセージを送信する．さらに，このメッセージ送信中に攻撃メッセージを送信することで，正規メッセージと攻撃メッセージがそれぞれ，正規の送信 ECU と攻撃者 ECU のバッファに溜める．すなわち，内容に意味を持たないメッセージ送信完了直後にこれらのメッセージが同時に送信を開始する．バスオフ攻撃は正規の送信 ECU の状態を遷移させる攻撃であり，なりすまし攻撃ではない．

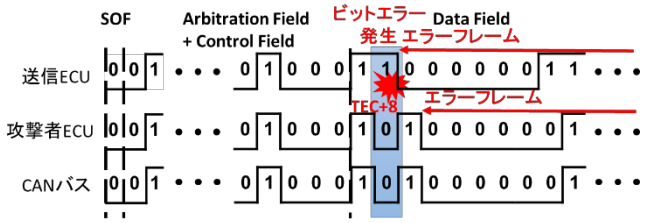


図 3 ビットエラーを用いたバスオフ攻撃

3. 提案方式

3.1 送信 ECU における不正送信阻止を回避するなりすまし攻撃方式

本節では，不正送信阻止方式を回避する電気的改ざんを用いないなりすまし攻撃を提案する．初めに，攻撃対象とする不正送信阻止方式[3][4]を実装した ECU が正しく動作する条件を整理する．不正送信阻止が正しく動作するのは，r0 ビットを受信したタイミングで正規の送信 ECU が送信中でない場合に限る．そのため，r0 ビット受信後に意図的に正規の送信 ECU の送信が停止され，引き続く形でなりすましメッセージが送信されれば，不正送信を阻止することができない．

正規の送信 ECU の送信を停止する方法を検討するうえで，エラー管理方式に着目した．CAN ではデータフレーム送信時にエラーが発生したとき，送信を中断しエラーフレームを送信する．送信されるエラーフレームは CAN コントローラの状態に応じて変わる．特に，エラーパッシブ状態ではエラーフラグが 6bit のリセッシブとなり，エラーフレームが 14bit のリセッシブで構成されるため，他の ECU からは送信を停止したと判断される．

正規の送信 ECU の送信を停止させたのち，引き続く形で改ざんされたデータフレームを送信する方式を検討する

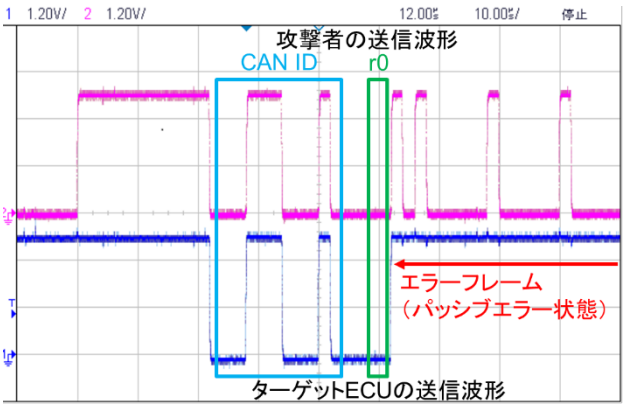


図 4 バスオフ攻撃による状態遷移時の攻撃波形

うえで，バスオフ攻撃が有効であると考えた．バスオフ攻撃では，正規の送信 ECU をパッシブエラー状態に遷移させ，同時に攻撃に用いたメッセージが送信される．さらに，メッセージが送信されている時を表す図 4 の波形図より，r0 ビットを送信するタイミングでは正規の送信 ECU と攻撃者 ECU が送信中であることが分かる．そのため，不正送信阻止方式では攻撃メッセージの送信を阻止することができない．

これより，条件（A）を満たす状況下で，バスオフ攻撃に利用される攻撃メッセージの制約である 2.5 節の条件 1，2 を満たすなりすましメッセージを正規メッセージと同時に送信することで，不正送信阻止方式を回避するなりすまし攻撃が可能となる．

3.2 ビットエラー発生時の ECU の挙動

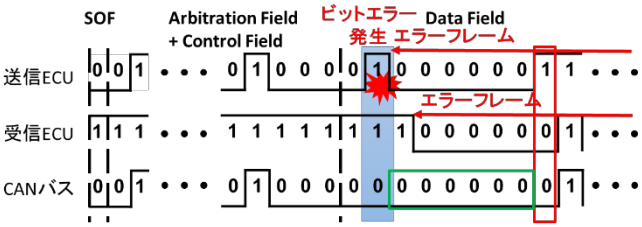
3.1 節で述べた攻撃方式では，既存の不正送信阻止方式がビットエラー発生時の挙動を十分に考慮できていないことを利用していた．そこで，この攻撃方式を防ぐ不正送信阻止方式を検討する前に，ビットエラー発生時の挙動について確認する．初めに，ビットエラー発生時における条件の組み合わせについて考える．ビットエラーはフレーム送信中に発生するため，送信 ECU の状態はエラーアクティブ状態かエラーパッシブ状態である．ビットエラーは送信ビットと受信ビット（CAN バス上の値）が異なることにより発生するため，送信ビットと受信ビットの値の組み合わせ[送信ビット，受信ビット]は[ドミナント，リセッシブ]と[リセッシブ，ドミナント]の 2 種類となる．これらの 2 条件の組み合わせを表 2 に示す．表 2 に示した 4 つの組み合わせにおける ECU の挙動を図 5 に示す．ここで，青背景がビットエラー発生時，赤枠がビットエラー発生後に最初に送信ビットと受信ビットが異なるビット，緑枠が赤枠の直前 6bit 期間中の受信ビットを表す．表 2 の全ての組み合わせにおいて，送信 ECU はビットエラー検出後エラーフレームを送信する．さらに，受信 ECU ではスタッフエラーを検出し，エラーフレームを送信する．

A，B の組み合わせについて，スタッフエラー検出タイミングについて詳細に確認する．ビットエラー発生直前にビットエラー発生時の受信ビットと同じ値が X bit 続いていたとすると，ビットエラー発生後 5-X bit 期間後にスタッフエラーを検出する．CAN ではスタッフィングルールにより同一ビットが 6bit 続くことはないため，X は最大 5 である．図 5（a）に X が 4 であるときの例を示す．X が 4 以下のとき，ビットエラー発生後 7bit 目の送信 ECU はエラーデリミタ（リセッシブ）を送信しており，受信 ECU がエラーフラグ（ドミナント）を送信中であるため，このビットがビットエラー発生後最初に送信ビットと受信ビットが異なるビットとなる．このビットの直前 6bit は送信 ECU が送信したエラーフラグとなるため，直前 6bit の受信ビットは全て同じ値をとる．なお，X が 5 であるとき，送信 ECU がビットエラーを検出すると同時に受信 ECU はスタッフエラーを検出し，送信 ECU と受信 ECU が同時にエラーフレームを送信開始する．そのため，ビットエラー発生後に送信ビットと受信ビットが異なることがない．

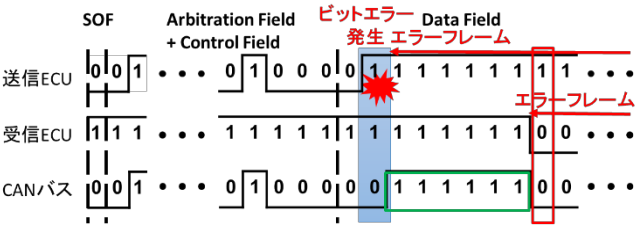
C の組み合わせについて，スタッフエラー検出タイミングについて詳細に確認する．ビットエラー発生直前にドミナントが X bit 続いていたとする．図 5（b）に X が 4 であるときの例を示す．X が 4 以下のとき，ビットエラー発生時の受信ビットはドミナントであり，ビットエラー発生直後はリセッシブを受信するため，スタッフエラーを検出するのは，送信 ECU がエラーフラグを送信し終えた時である．この直後に受信 ECU がエラーフレームを送信開始し，ビットエラー発生後に初めて送信ビットと受信ビットが異なる．ここで，直前 6bit は送信 ECU が送信したエラーフラグとなるため，直前 6bit の受信ビットは全て同じ値をとる．X が 5 であるとき，送信 ECU がビットエラーを検出すると同時に受信 ECU はスタッフエラーを検出し，送信 ECU

表 2 ビットエラー発生時における条件の組み合わせ

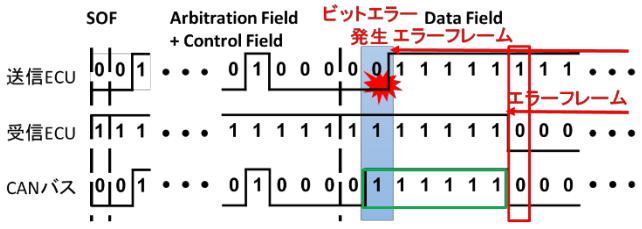
		送信ビットと受信ビットの組み合わせ	
		送信ビット：1 受信ビット：0	送信ビット：0 受信ビット：1
エラー発生時における ECU の状態	エラーアクティブ	A	B
	エラーパッシブ	C	D



(a) A,B の組み合わせにおける挙動（X=4）



(b) C の組み合わせにおける挙動（X=4）



(c) D の組み合わせにおける挙動（X=0）

図 5 各組み合わせにおけるビットエラー発生時の挙動

と受信 ECU が同時にエラーフレームを送信開始する．ここで，送信 ECU はパッシブエラーフレームを送信し，受信 ECU はアクティブエラーフレームを送信するため，エラーフレーム送信開始時に送信ビットと受信ビットが異なる．この時，受信 ECU がスタッフエラーを検出していることから，直前 6bit の受信ビットが全て同じ値であることが分かる．

D の組み合わせについて，スタッフエラー検出タイミングについて詳細に確認する．ビットエラー発生直前にリセッシブが X bit 続いていたとする．図 5（c）に X が 0 であるときの例を示す．ビットエラー発生後 5-X bit 期間後にスタッフエラーを検出する．この時，送信 ECU はパッシブエラーフレームを送信しており，受信 ECU がアクティブエラーフレームを送信開始するので，ビットエラー発生後，初めて送信ビットと受信ビットが異なる．この時，受信 ECU がスタッフエラーを検出していることから，直前 6bit の受信ビットが全て同じ値であることが分かる．

全てのパターンに共通する特徴として，ビットエラー発生後，最初に送信ビットと受信ビットが異なるビットの直前 6bit の受信ビットが全て同じ値になることが挙げられる．

3.3 提案攻撃方式に対する不正送信阻止方式

提案攻撃方式では，既存の不正送信阻止方式は正規の送信 ECU がエラーパッシブ状態かつビットエラーが発生したときの挙動を十分に考慮していないことを利用していた．本論文では，ビットエラー発生時の挙動を十分に考慮した不正送信阻止方式を提案する．

なりすまし攻撃を行う攻撃者 ECU の送信ビットについて考える．攻撃者は CAN の仕様に従っているとは限らない．しかし，正規の受信 ECU でなりすましメッセージを受けとれる必要があるため，送信するデータフレームの構成は CAN の仕様に従う必要がある．ビットスタッフィングルールについても同様のことが言えるため，攻撃者 ECU の送信ビットが 6bit 期間連続して同じ値を取り続けることはない．

3.2 節より，正常時はビットエラー発生後，最初に送信ビットと受信ビットが異なるビットの直前 6bit の受信ビットが全て同じ値となる．また，なりすまし攻撃時は 6bit 期間連続して同じ値を取り続けることはない．これより，ビットエラー発生後，最初に送信ビットと受信ビットが異なるビットの直前 6bit の値をサンプリングすることで，なりすまし攻撃を検知できることが分かる．ここでは，正規のメッセージとなりすましメッセージが同時に送信された場合についてのみ検討した．また，これらのメッセージが同時に送信されなかった場合は，既存の不正送信阻止方式で対応可能である．そこで，既存の不正送信阻止方式を改良することで，提案攻撃方式をも防げる不正送信阻止方式の具体的な方法を次に示す．

- SOF 受信時または送信時
  1. フラグを下げる
- r0 ビット受信時または送信時
  1. 受信した CAN ID が自身の持つ CAN ID と等しく，送信中であれば，フラグを立てる．
  2. 受信した CAN ID が自身の持つ CAN ID と等しく，送信中でなければ，アクティブエラーフレームを送信する．
- ビットエラー検出後，最初に送信ビットと受信ビットが異なる時
  1. フラグが立っていて，直前 6bit の値全てが同じでなければ，アクティブエラーフレームを送信する．

提案方式では，ビットエラー発生後に送信ビットと受信ビットが異なるビットが出現することを前提としている．しかし，表 2 の A，B の組み合わせにおいて，ビットエラー発生直前にビットエラー発生時の受信ビットと同じ値が 5bit 続いていたとき，送信ビットと受信ビットが異なるビットが出現することなく，エラーフレームの送信を終えるが，正規の送信 ECU と受信 ECU がアクティブエラーフレームを送信していることから，なりすましメッセージが送信される余地はない．そのため，このことは問題とならない．

3.4 提案方式の実装

提案方式の実現に必要な機能を以下に示す．

a. 送信 ECU の送信ビットと受信ビットのサンプリング
b. 6bit 期間の送信ビットを保存できるメモリ
c. SOF 検出機能
d. アクティブエラーフレーム送信

これらの機能はもちろん CAN コントローラを改造することでも実現可能である．しかし，ここではより多くの場合に適用できるように，CAN コントローラを改造せずに拡張する形で実現する方法を検討する．a. の機能は CAN コントローラと CAN トランシーバの間の通信をサンプリングすることで実現可能である．b. の機能はフリップフロップを使うことで実現可能である．c. の機能は a. と b. の機能を組み合わせことで実現可能である．d. の機能は CAN コントローラと CAN トランシーバを共有することで実現可能である．図 6 に提案方式の実装例を示す．

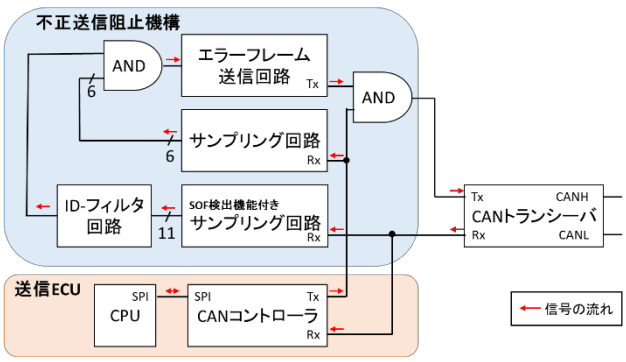


図 6 提案方式の実装例

4. 評価実験

4.1 実験環境

提案攻撃方式の実現可能性を確認するために行う実験に用いた CAN バスの模擬環境を図 7 に示す．模擬環境は従来の不正送信阻止方式を実装した送信 ECU と攻撃者 ECU，受信 ECU と CAN バスの状態を確認するために用いる CAN アナライザとオシロスコープから構成される．利用した正規メッセージと攻撃メッセージを表 3 に示す．攻撃メッセージの CAN ID は正規メッセージと同一のもので，DLC とペイロードは 2.5 節の条件 1,2 を満たすものとした．

提案不正送信阻止方式の実現可能性を確認するための実験でも，提案攻撃方式の実現可能性を確認する際に利用した模擬環境を用いる．ただし，送信 ECU は提案方式を実装したものを用いる．

従来方式の実装について，文献[4]では CAN コントローラ上に実装を行っていたが，今回は CAN コントローラを拡張する形で実装を行った．

4.2 攻撃方式の実験結果

提案攻撃方式を行った結果を図 8 と図 9 に示す．青枠に囲まれているメッセージが正規メッセージで，赤枠がなりすましメッセージ，緑枠が同時送信させるために用いた内容に意味をもたないメッセージである．図 9 より同時に送信を開始し 16 回再送を繰り返していることが分かる．図 10 の CAN Bus Event に着目すると，REC が 1 ずつ増加していることが分かる．これは，受信中のメッセージがエラーになったことを表す．REC が 1 ずつ増加し，16 まで増加していることから再送が送信 ECU と攻撃者 ECU の送信したデータフレームが 16 回エラーになっていることが分かる．すなわち，送信 ECU と攻撃者 ECU がエラーパッシブ状態に遷移した．送信 ECU がエラーパッシブ状態に遷移した後，なりすましメッセージが注入されている．これより，提案攻撃方式の実現可能性が確認できた．

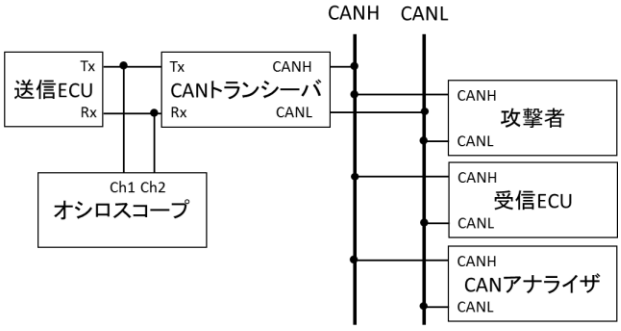


図 7 CAN バスの模擬環境

表 3 実験に用いたメッセージ

正規メッセージ	CAN ID: 1C4 （周期 23ms） Data: 00 00 00 00 00 00 00 00
なりすましメッセージ	CAN ID: 1C4 （周期 23ms） Data: 17 70 00 00 00 00 00 00

4.3 不正送信阻止方式の実験結果

提案した不正送信阻止方式を実装した送信 ECU に対して，提案攻撃方式を行ったときの結果を図 10 に示す．青枠に囲まれているメッセージが正規メッセージで，赤枠がなりすましメッセージ，緑枠が同時送信させるために用いた内容に意味をもたないメッセージで，黄枠が送信阻止されたなりすましメッセージである．ここで，提案方式で阻止可能にした赤枠の部分について詳細を確認する．図 10 の

Time (abs/rel)	Description	ArbId/Header	Len	DataBytes
	HS CAN $1C4	1C4	8	FF FF FF FF FF FF FF FF
22.983 ms	HS CAN $0	0	8	00 00 00 00 00 00 00 00
73 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 1	3	00 01 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 2	3	00 02 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 3	3	00 03 00
111 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 4	3	00 04 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 5	3	00 05 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 6	3	00 06 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 7	3	00 07 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 9	3	00 09 00
87 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 10	3	00 0A 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 11	3	00 0B 00
84 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 12	3	00 0C 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 13	3	00 0D 00
84 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 14	3	00 0E 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 15	3	00 0F 00
85 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 16	3	00 10 00
258 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 15	3	00 0F 00
3 µs	HS CAN $1C4	1C4	8	00 00 00 00 00 00 00 00
275 µs	CAN Bus Event	CAN Rx/Tx REGS - TEC: 0 - REC: 14	3	00 0E 00
3 µs	HS CAN $1C4	1C4	8	FF FF FF FF FF FF FF FF
21.461 ms	HS CAN $1C4	1C4	8	FF FF FF FF FF FF FF FF

図 8 バスオフ攻撃による状態遷移時の攻撃波形

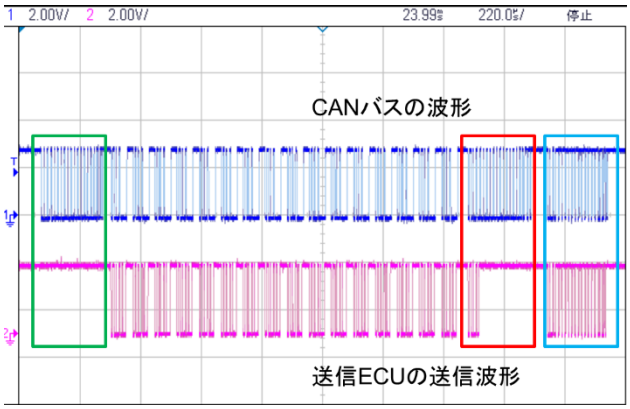


図 9 バスオフ攻撃による状態遷移時の攻撃波形

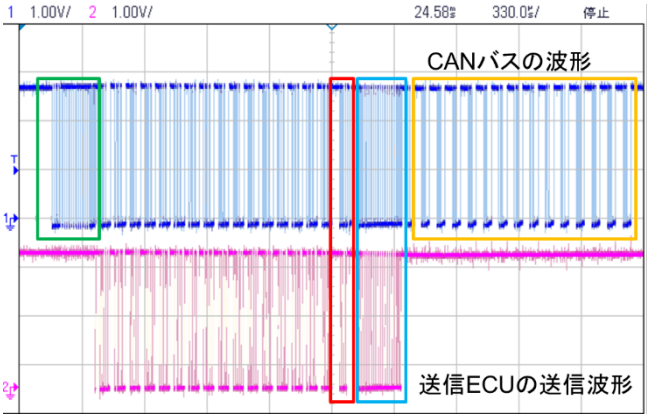


図 10 バスオフ攻撃による状態遷移時の攻撃波形

赤枠の部分を拡大した図を図 11 に示す. 青枠はビットエラー発生時，赤枠はビットエラー発生後最初に送信ビットと受信ビットが異なる時, 緑枠は赤枠の直前 6bit 期間を表す. 緑枠の間全ての値が同じではないため，赤枠の次からアクティブエラーフレームが送信され，なりすましメッセージの送信が阻止されている．また，CAN アナライザで受信結果を確認したところ，なりすましメッセージが観測されることはなかった．これより，提案攻撃方式も阻止することができる不正送信阻止方式が実現できたと言える．

5. 考察

本章では, 先行研究である既存の不正送信阻止方式[3][4]と周期に着目した防御方式[6]と提案した不正送信阻止方式における攻撃に対しての効果を比較する．効果の指標として，検知率と検知後阻止可能であるかを比較する．対象とする攻撃として，電気的改ざんを用いない方式である単純になりすましメッセージを注入する攻撃[1][2]と提案した攻撃方式を用いる．また，電気的改ざんを用いる方式として，周期なりすまし攻撃[9]と強いリセッシブを用いたエラーフレームに上書きされないメッセージを利用した攻撃[11]とサンプルポイントのずれを利用した攻撃[10]を考える．防御方式の効果についてまとめたものを表 4 に示す.

周期性に着目した方式では，単純ななりすまし攻撃に対して 100%の検知が可能であると報告されている. ただし，提案した攻撃方式と周期なりすまし攻撃とサンプルポイントのずれを利用した攻撃では周期もなりすますため，検知率が大幅に下がることが推測される．既存の不正送信阻止方式では，正規メッセージの送信時以外に送信されたなりすましメッセージは確実に検知することが可能である．そのため，提案した攻撃方式とサンプルポイントのずれを利用した攻撃以外は確実に検知可能である．提案した不正送信阻止方式では，既存の不正送信阻止方式で検知可能な項目に追加して同時に送信された場合にも検知可能である．ただし，サンプルポイントのずれを利用した攻撃では，正規の送信 ECU から送信されたビットを改ざんしているため，検知できない．全ての防御方式において，強いリセッシブ利用したフレームの送信を阻止することはできない.

提案した不正送信阻止方式では，強いリセッシブ利用したフレームの送信を阻止することはできず，サンプルポイントのずれを利用した攻撃については検知することさえもできない．そのため，電気的改ざんを用いた攻撃に対しては脆弱である．しかし，電気的改ざんを用いない攻撃は確実に検知し，阻止することが可能である．そのため，提案した不正送信阻止方式は電気的改ざんを行われないことが約束された環境では提案した不正送信阻止方式が有効であると言える.

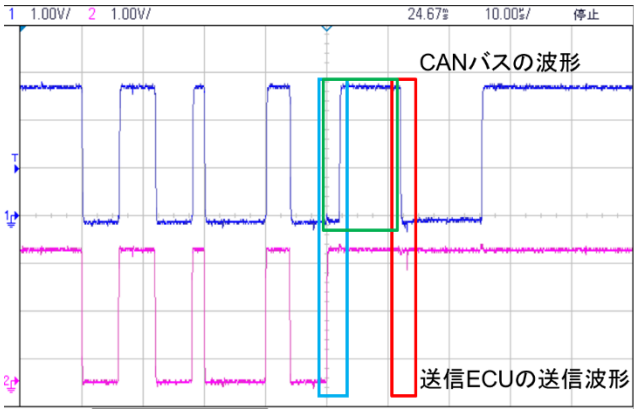


図 11 バスオフ攻撃による状態遷移時の攻撃波形

6. おわりに

本論文では，送信 ECU における不正送信阻止方式に対して電気的改ざんを用いない攻撃方式を提案し，模擬環境で評価実験を行うことにより実現可能であることを確認した．さらに，提案攻撃方式も阻止することができる不正送信阻止方式の提案を行い, その有効性を確認した. しかし，提案した不正送信阻止方式では，電気的改ざんを用いた攻撃に対して脆弱であることが分かっている．今後の課題として，電気的改ざんを用いた攻撃も防ぐことができる不正送信阻止方式の検討を行う.

謝辞

本研究の一部は，広島市立大学特定研究費により行われた．ここに記して謝意を表す.

参考文献

[1] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage: Experimental Security Analysis of a Modern Automobile, Proc. 2010 IEEE Symposium on Security and

表 4 防御方式の効果

防御方式	電気的改ざんを用いない攻撃				電気的改ざんを用いた攻撃					
	単純なりすまし攻撃		提案した攻撃方式		周期なりすまし攻撃		強いリセッシブを利用したフレーム		サンプルポイントのずれを利用した攻撃	
	検知	阻止	検知	阻止	検知	阻止	検知	阻止	検知	阻止
周期性を用いた方式	△	△	×	×	×	×	○	×	×	×
既存の不正送信阻止方式	○	○	×	×	○	○	○	×	×	×
提案した不正送信阻止方式	○	○	○	○	○	○	○	×	×	×

Privacy, pp.447-462, May 2010.
[2] Takaya Ezaki, Tomohiro Date, and Hiroyuki Inoue: An Analysis Platform for the Information Security of In-vehicle Networks Connected with the External Networks, Proc. The 10th International Workshop on Security, pp.301-315, Aug. 2015.
[3] 畑正人，田邉正人，吉岡克成，大石和臣，松本勉: 不正送信阻止：CAN ではそれが可能である, CSS2011, pp.624-629, Oct. 2011.
[4] 畑 正人, 田邉 正人, 吉岡 克成, 松本 勉: CAN における不正送信阻止方式の実装と評価, 電子情報通信学会技術研究報告, vol.112, no.342, pp15-22, Dec. 2012.
[5] International Organization for Standardization: Road vehicles, controller area network (CAN), Part 1: Data link layer and physical signaling, ISO IS11898-1, 2003.
[6] 倉地 亮, 高田 広章, 上田 浩史, 堀端 啓史: 車載制御ネットワークにおける送信周期監視システムの提案, SCIS2015, pp.1-7, Jan. 2015.
[7] A. Herrewege, D. Singelee, and I. Verbauwhede: CANAuth-a simple, backward compatible broadcast authentication protocol for CAN bus, 9th Embedded Security in Cars Conf., Sep. 2011.
[8] 中野 将志, 久保田 貴也, 汐崎 充, 藤野 毅: 車載 CAN 通信の暗号化とリプレイ攻撃対策手法の実装評価, 研究報告組込みシステム（EMB）, pp.1-6, Feb. 2015.
[9] 家平和輝，井上博之，石田賢治: 特定の CAN メッセージを送信する ECU に対するバスオフ攻撃を利用したなりすまし攻撃, DICOMO2017, pp.1163-1168, June 2017.
[10] 松本勉, 中山淑文, 向達泰希, 土屋遊, 吉岡克成: CAN における再同期を利用した電気的データ改ざん, SCIS2015, pp.1-8, Jan. 2015.
[11] 菅原健，佐伯稔，三澤学: 強いリセッシブを用いた CAN の電気的データ改ざん, 電子情報通信学会技術研究報告 ICSS, vol.114, no.489, pp.67-72, Feb. 2015.
[12] 小林 優希，中山 淑文，松本勉: CAN における不正送信阻止が可能となる条件, CSS2015, pp.116-123, Oct. 2015.
[13] Cho, Kyong-Tak and Shin, Kang G.: Error Handling of In-vehicle Networks Makes Them Vulnerable, Proc. 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS2016), pp.1044-1055, Oct. 2016.