車載ネットワークCANにおけるバスオフカウンター攻撃の改善

相馬 大輔[1]　森 彰[1]　山本 秀樹[2]　畑 洋一[2]

概要：近年、自動車は複雑化・ネットワーク化が進んでいる。これに伴い、様々なサイバー攻撃手法が報告されている。特に、2016 年には Cho、Shin によってバスオフ攻撃と呼ばれる新しい DoS 攻撃の手法が報告されている。バスオフ攻撃は Control Area Network (CAN) の仕様を悪用し、特定のノードをサービス不能にする攻撃である。報告では攻撃手法のみでなく、サービス不能状態を防ぐ対策も報告されている。しかし、報告された対策は攻撃による悪影響を排除することはできなかった。筆者らは 2018 年に攻撃を行うノードをバスから排除する対抗策を提案した。提案した対抗策は攻撃者に対しカウンター攻撃を行う事でサービス不能にする。この対抗策にも、カウンター攻撃実行が可能になるまで数回の攻撃を許容しなくてはならないという課題がある。本論文では、この課題を改善し即時カウンター攻撃を実行可能とする方法について検討状況を報告する。

キーワード：車載ネットワーク, Control Area Network (CAN), バスオフ攻撃, 対抗策

Improvement of the counter bus-off attack

Daisuke Souma[1]　Akira Mori[1]　Hideki Yamamoto[2]　Yoichi Hata[2]

Abstract: Recent automotive systems are increasingly complex and networked. The situation has given rise to various cyber-attack methods. Cho and Shin introduced a new type of Denial of Service (DoS) attacks called bus-off attacks, which abuses certain properties of Control Area Network (CAN) used for vehicle control. They not only introduced a novel software based attack method but also proposed a countermeasure which resets the victim node to keep it from going into the disabled state. However, their countermeasure could not avoid unintended effects caused by the attack. The authors proposed a countermeasure to avoid unintended effects. It forces the node that started the bus-off attack into the disabled state in a way similar to the original bus-off attack. But it required some interval from the attack detection to the counter attack. In this paper we report research status to improve the countermeasure.

Keywords: In-vehicle Network, Control Area Network (CAN), Bus-off Attack, Countermeasure

1. はじめに

近年の自動車システムは複雑化、ネットワーク化が進む事で、周辺と協調した高度な制御が可能になっている。しかし、これらによって意図しない影響ももたらされることになる。特にネットワーク化は、自動車への攻撃者の侵入機会を増やし、制御を奪うような攻撃をリモートで可能としてしまう。実際、リモートでの制御を奪う攻撃事例が研究者により複数報告されている。特に、Miller と Valasec による報告 [15] は、140 万台のリコールを引き起こした。

車両の制御を奪うためには車載ネットワークに関しての理解が不可欠である。CAN (Contorl Area Network) は車両制御のための通信に使用される最も重要な車載ネットワークの一つである。しかし、CAN はセキュリティや

[1] 国立研究開発法人 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology
[2] 住友電気工業株式会社
Sumitomo Electric Industries, Ltd.

© 2018 Information Processing Society of Japan

認証に関する機能を持たないため、接続するだけで通信の解析、メッセージのインジェクションなどの攻撃が実行できる。そのような攻撃を防ぐために、メッセージ認証コード (MAC) や暗号化といった方法が提案されている [6], [10], [18], [24]。

2016 年 Cho と Shin らによりバスオフ攻撃と呼ばれる新しい DoS 攻撃が報告された [2]。これは、CAN のエラー処理メカニズムを悪用し、対象がメッセージの送受信をできない状態にするものである。バスオフ攻撃では CAN の仕様に従った正常なメッセージを攻撃に用いており、MAC や暗号化などの方法で防ぐことが困難である。報告では攻撃方法のみでなく対抗策も提案しているが、対象ノードをリセットしバスオフを回避するというものであり、攻撃の影響は排除できないものであった。

2018 年に筆者らはバスオフ攻撃の影響を排除する対抗策を提案した。提案した対抗策はバスオフ攻撃を行うノード (攻撃者) に対しカウンター攻撃を行い、攻撃者をバスオフ状態にするものである。そのため、対抗策により攻撃の悪影響を排除することが可能である。一方で、この対抗策にはいくつかの弱点があることがわかっている。「自然に連続して発生したエラーを、バスオフ攻撃として誤検知してしまう。」、「対抗策の原理を知ることで、カウンター攻撃を回避できる。」、「検知からカウンター攻撃までインターバルが必要である。」などである。

本論文では、これらの弱点のうち「検知からカウンター攻撃までインターバルが必要である。」に対する改善について、その検討状況を報告する。

本論の構成は以下の通りである。2 節は想定するバスオフ攻撃の攻撃シナリオについて説明する。3 節では準備として本論文に関係する CAN の仕様、バスオフ攻撃、提案済みの対抗策について説明する。4 説では対抗策の改善について説明し、そのフィージビリティを確認するための実験結果および考察を述べる。最後に 5 説で検討状況をまとめ、今後の方針を述べる。

2. 準備

本節では準備として、本論文に関連する CAN の仕様、バスオフ攻撃の仕組み、提案されているカウンター攻撃による対抗策について説明する。

2.1 CAN の概要

CAN は車載ネットワークで広く使用されている、マルチマスタ、シリアルバスプロトコルで、Bosch 社によって開発された [7]。ノードは CAN high、CAN low と呼ばれるツイストペアで接続され、その電位差を信号として使用している。2 線間の 2V の電位差を “0” (ドミナント)、0V の電位差を “1” (リセッシブ) と定義している。つまり信号は非対称であり、ドミナントがリセッシブを上書きできる。以下でより詳細な仕様について説明する。

2.1.1 フレーム (メッセージ)

CAN で使用されるフレーム (メッセージ) は以下の 4 つである。

- データフレーム
- リモートフレーム
- エラーフレーム
- オーバーロードフレーム

ここでは、本論文に関連するデータフレームとエラーフレームについてのみ説明する。

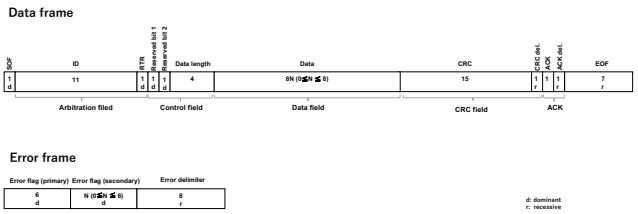

図 1 Data frame and Error frame

データフレームはデータを送信するためのフレームである (図 1)。データフレームは一度に最大 8 byte のデータを送信できる。また、ID はメッセージ送信の調停 (アービトレーション) や受信ノードでのフィルタリングに使用される。

エラーフレームは通信でエラーが発生したことを知らせるためのフレームである。図 1 で示しているように、primary error flag、secondary error flag、error delimiter で構成される。エラーを検知したノードは primary error flag として 6 ビットの連続したドミナントを送信する。primary error flag は bit stffing rule(2.1.3 節で説明) を侵害し、その他全てのノードでもエラーが検出され secondary error flag(6 ビットの連続したドミナント) を送信する。その後 error delimiter が送信され、通常状態へ復帰する。

各フレームは Inter Frame Space (IFS) と呼ばれるリセッシブ 3 ビットで分割される。

2.1.2 Arbitration

次にノードのメッセージ送信権獲得について説明する。CAN バスがアイドル状態の場合、全てのノードがメッセージを送信可能である。あるノードが送信を開始すると、そのメッセージの送信完了まで送信待機する。

二つ以上のノードが同時にメッセージ送信を開始した場合、アービトレーションメカニズムによりメッセージ送信の調停が行われる。各ノードは自身の送信信号と実際のバスレベルを比較し、一致する場合は送信を続けるが、異なる場合は調停に負けたと判断し送信を取りやめる。アービトレーションフィールドの最後まで送信し続けたノードが調停に勝ち、送信権を獲得する。つまり若い ID の方が高い優先度を持っている。

© 2018 Information Processing Society of Japan

2.1.3 Error handling

CAN では以下の 5 種類のエラーが定義されている。

- ビットエラー: アービトレーションフィールド以外で、送信した信号と実際のバスレベルが異なる場合のエラー。
- スタッフエラー: ビットスタッフィングルールを逸脱した場合のエラー。ビットスタッフィングルールとは、同じ信号が 5 ビット続いた場合にその逆の信号を 1 ビット挿入するというものである。
- CRC エラー: 計算で得られる CRC と受信した CRC が異なる場合のエラー。
- フォームエラー: ACK デリミタや EOF などリセッシブである部分でドミナントを受信した場合のエラー。
- ACK エラー: 送信ノードが ACK スロットでドミナントを受信できなかった場合のエラー。

上記エラーのうちビットエラーと ACK エラーは送信ノードのエラーであり、その他のエラーは受信ノードのエラーである。

CAN にはエラーを頻発するノードのネットワークへの影響を排除するためのメカニズムがある。各ノードは Transmit Error Counter (TEC) と Receive Error Counter (REC) という二つのエラーカウンターを持っている。それらのカウンターは次のルールに従って増減する。送信ノードのエラーでは、送信ノードの TEC は 8 増加し、その他のノードの REC は 1 増加する。受信ノードのエラーでは、受信ノードの REC が 8 増加する。メッセージの送信が問題なく完了した場合、TEC と REC 双方の値が 1 減少する。

各ノードは、その TEC と REC の値により状態が変化する (図 2)。エラーアクティブ状態は特に制限のない通常状態である ($TEC \leq 127$ and $REC \leq 127$)。エラーパッシブ状態はエラーが頻発しているため、通信が制限されている状態である ($127 < TEC < 256$ or $127 < REC < 256$)。エラーパッシブ状態では連続してメッセージを送信する場合、メッセージ送信のために必要な IFS が通常より 8 ビット必要となる (パッシブ IFS)。さらに、エラー発生時に他の通信に影響を及ぼさないために、エラーフラグが通常のドミナント 6 ビットからリセッシブ 6 ビットに変化する。

エラーパッシブ状態のノードがさらにエラーを頻発すると、ノードはバスオフ状態へと遷移する ($TEC \geq 256$)。バスオフ状態で、ノードはロジカルにバスから切り離され、メッセージの送信が不可能になる。

2.2 バスオフ攻撃とその対抗策

2.2.1 バスオフ攻撃

バスオフ攻撃は CAN のエラー処理メカニズムを悪用した DoS 攻撃である。攻撃を行うノード (攻撃者) は攻撃の対象ノード (被害者) に送信エラーを繰り返し起こさせ、バスオフ状態にすることでサービスを不能にする。

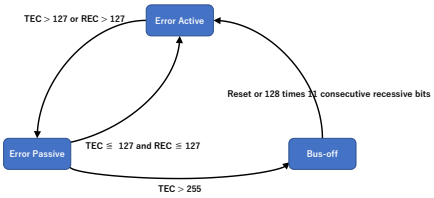

図 2 State diagram of CAN nodes

バスオフ攻撃は 2 種類の攻撃手法が報告されている。一つ目の攻撃手法は CAN メッセージを攻撃として使用するものである [2]。攻撃に使用するメッセージ (攻撃メッセージ) は被害者の送信するメッセージ (対象メッセージ) と ID は同じであるが異なるデータを持っている。この攻撃メッセージを対象メッセージと全く同じタイミングで送信することで、ビットエラーを引き起こすことができる。繰り返しビットエラーを起こすことで、被害者の TEC は増加し最終的に被害者はバスオフ状態へと遷移する。この攻撃手法は 2 つのフェイズに分かれる。フェイズ 1 は攻撃者、被害者共にエラーアクティブ状態である。攻撃が開始されると両者でビットエラーが起こり、TEC が 8 増加する。ビットエラーに対するエラーフレーム送信完了後、両者は再送信を同時に試みるため、自動的に再度ビットエラーが起こる。これが 16 回連続して起こり、両者はエラーパッシブ状態 ($TEC = 128$) へ遷移す。フェイズ 2 は少なくとも一方のノードがエラーパッシブ状態である。フェイズ 2 では、まず攻撃により被害者のみがビットエラーを起こし TEC が 8 増加する。この際、被害者はエラーパッシブ状態であるため、攻撃者はそのビットエラー発生に気づかずに送信を完了し、TEC が 1 減少する。攻撃メッセージ送信完了後、被害者はメッセージ再送信を行い TEC が 1 減少する。結果として一度の攻撃で被害者の TEC は 7 増加し、攻撃者の TEC は 1 減少する。これが 19 回繰り返すことで、被害者はバスオフ状態へ遷移する ($TEC = 261$)。

二つ目の攻撃手法は CAN のプロトコルに従わない信号列を攻撃として使用する [9], [19]。攻撃者は実際のバスレベルをビット毎に監視する。対象メッセージの ID が検知されると、6 ビット以上の連続したドミナント (攻撃信号) を送信する。ビットスタッフィングルールから、この攻撃信号により被害者はビットエラーを起こし、TEC が 8 増加する。これを 32 回繰り返すことで、被害者はバスオフ状態 ($TEC = 256$) へと遷移する。攻撃信号の長さにより一度の攻撃で増加する被害者の TEC は変化する。大量の連続ドミナントを送信すると、初めの 6 ビットがビットエラーを引き起こし TEC を 8 増加させる。続く 14 ビットがエラーを引き起こし、さらに TEC を 8 増加させる。その後 8 ビット毎にエラーが起き、TEC が 8 づつ増加する。そのため、260 ビットを超える連続ドミナントの送信を一

© 2018 Information Processing Society of Japan

度行うだけで被害者をバスオフ状態へと遷移させられる。

本論文で提案する対抗策が対象としているのは前者の CAN メッセージを攻撃に使用するバスオフ攻撃である。一方で、後者の攻撃信号を使用したバスオフ攻撃は、攻撃者をバスオフ状態へ遷移させるためのカウンター攻撃として使用する。

2.2.2 カウンター攻撃による対抗策

本節では既に提案しているカウンター攻撃を用いた対抗策について説明する。提案されている対抗策はバスオフ攻撃の検知と攻撃者へのカウンター攻撃で構成される。

バスオフ攻撃の検知はフェイズ 1 で現れる信号を用いる。この検知手法は Cho と Shin により提案されている方法の一部を使用している [2]。フェイズ 1 では全く同じビットエラーが連続して発生する。このようなエラーが発生する確率は非常に低く、偶発的に発生したとは考えにくい。そこで、全く同じエラーが連続して発生した場合、バスオフ攻撃が行われていると判断する。

次にカウンター攻撃について説明する。カウンター攻撃には亀岡らにより提案された大量のドミナント送信によるバスオフ攻撃を用いる [9]。カウンター攻撃を攻撃者のみに作用させることができれば、被害者より先に攻撃者をバスオフ状態へと遷移できる。攻撃者のみにカウンター攻撃を行うためには、攻撃者のみがメッセージを送信している状態が必要である。バスオフ攻撃中は必ず攻撃者が被害者のメッセージを上書きしているため、自然にそのような状況は現れない。そこで、ノードの状態による IFS の違いを利用し攻撃者のみが送信している状況を作り出す。バスオフ攻撃のフェイズ 2 では、攻撃者と被害者は共にエラーパッシブ状態 ($TEC = 128$) である。その後、攻撃毎に攻撃者の TEC は 1 減少し、被害者の TEC は 7 増加する。9 回の攻撃が完了した段階で、攻撃者はエラーアクティブ状態 ($TEC = 119$)、被害者はエラーパッシブ状態 ($TEC = 191$) となる。フェイズ 2 の 10 回目の攻撃時にカウンター攻撃ノードから信号を送信し、攻撃者、被害者にビットエラーを起こす。このエラーにより両者はメッセージの再送を試みる。攻撃者はエラーアクティブ状態 ($TEC127$) で、エラーフレーム送信後、通常 3bit の IFS で再送を開始する。一方で被害者はエラーパッシブ状態 ($TEC = 199$) で、エラーフレーム送信後、11bit のパッシブ IFS で再送を開始する。これにより、攻撃者のみがメッセージを再送信し、被害者は送信待機状態になる。この攻撃メッセージに対しカウンター攻撃を行うことで、攻撃者のみをバスオフ状態へと遷移できる。

3. 対抗策の課題と改善の検討および考察

本節では、提案されている対抗策の課題と改善のための検討と、その実現可能性確認のために実施した実験の説明および考察を述べる。

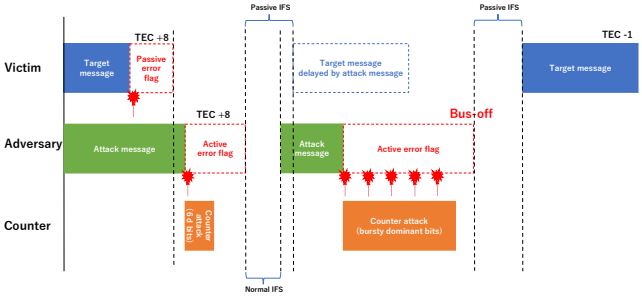

図 3 Counter attack method

3.1 対抗策の課題と改善の検討

提案済みの対抗策には以下の課題がある。

(1) バスオフ攻撃の誤検知
(2) カウンター攻撃を悪用可能
(3) 検知からカウンター攻撃までインターバルが必要

これらのうち、本論文では (3) に対する検討状況を報告する。

提案されている手法がインターバルを必要とするのは、カウンター攻撃実行のために必要な「攻撃者のみがメッセージを送信しているタイミング」を作り出すためである。

バスオフ攻撃検知後、カウンター攻撃実行までのインターバルを減らすために、フェイズ 2 における攻撃者と被害者の状態について検討する。被害者はビットエラーを検知後、パッシブエラーフラグを送信し送信待機状態になる。一方、攻撃者はメッセージの送信を継続する。被害者がパッシブエラーフラグ送信完了後で攻撃者が送信を継続している間は、攻撃者のみがメッセージ送信をしていると考えられる (図 4)。そのため、このタイミングでカウンター攻撃を行うと、攻撃者のみがバスオフ状態へ遷移すると考えた。

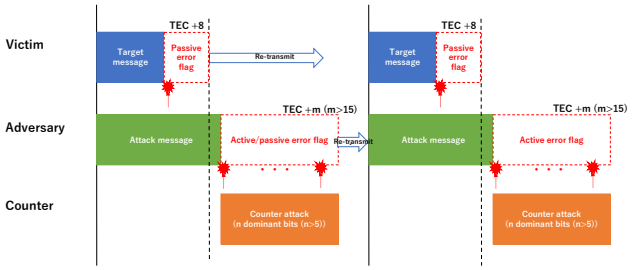

図 4 Counter attack timing

3.2 実験と考察

3.1 節のタイミングでのカウンター攻撃が可能であるかを検証するために、CAN シミュレータおよびカウンター攻撃デバイスにより実現可能性を確認した。実験で使用した CAN シミュレータおよびカウンター攻撃デバイスはラ

© 2018 Information Processing Society of Japan

ズベリーパイ、CAN コントローラ、CAN トランシーバで作成した。

実験の結果、対抗策によるカウンター攻撃の成功率は 30% 程度であった。カウンター攻撃の失敗時は必ず攻撃者と被害者の双方がバスオフ状態へ遷移した。カウンター攻撃が失敗した場合、成功した場合それぞれのバスレベルをオシロスコープでモニターしたのが図 5、6 である。

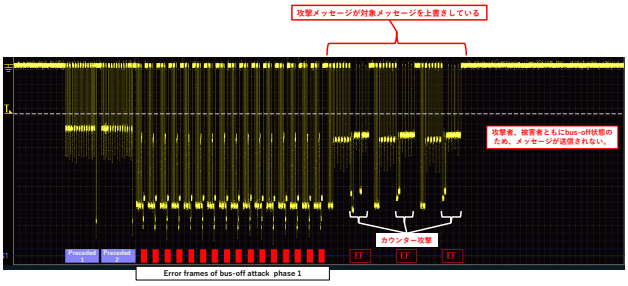

図 5 カウンター攻撃失敗時のバスレベル

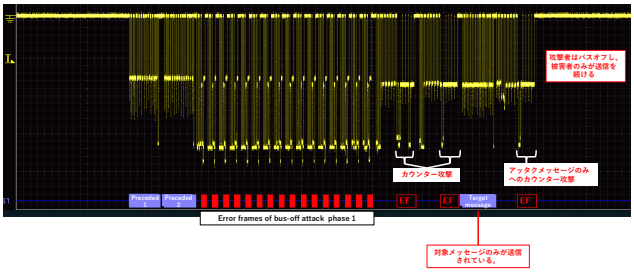

図 6 カウンター攻撃成功時のバスレベル

図 5 から、カウンター攻撃は想定したタイミングで実行されていることがわかる。しかしカウンター攻撃完了後、攻撃者だけでなく被害者もバスオフ状態へ遷移していた。このことから攻撃者だけではなく送信待機状態と考えた被害者にもカウンター攻撃が作用しており、改善案で検討したタイミングは被害者も送信中と認識していると考えられる。

一方で成功した場合は、攻撃メッセージよる対象メッセージの上書きが行われず、対象メッセージの送信のみが行われている (図 6)。その後の送信された攻撃メッセージにカウンター攻撃が行われ、攻撃者のみがバスオフ状態へ遷移する。カウンター攻撃の影響により攻撃者と被害者の送信タイミングに差異が生じたため、このようなことが起きたいと考えられるが、その原因・条件は特定できていない。

4. まとめと今後の課題

本論文では、Cho と Shin により提案されたバスオフ攻撃に対し、攻撃者をバスオフ状態へ遷移させるカウンター攻撃による対抗策の改善案について、現状の検討状態を報告した。当初、想定していた改善案は実験により有効なタイミングではないことがわかった。一方で、そのタイミングでカウンター攻撃を実施することで、カウンター攻撃のタイミングが作り出せる可能性があることを発見した。このタイミングができるか原因・条件はは不明確であり、今後検討を行なう。また、本論文では検討できていない課題の改善方法の検討も実施する予定である。

参考文献

[1] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner and T. Kohno, Comprehensive Experimental Analyses of Automotive Attack Surfaces, 20th USENIX conference on Security, 2011.

[2] K. Cho, and K. G. Shin. Error handling of in-vehicle networks makes them vulnerable, Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2016.

[3] T. Dagan and A. Wool, Parrot, a software-only anti-spoofing defense system for the CAN bus, 5th Embedded Security in Cars (ESCAR Europe), 2016.

[4] T. Dagan and A. Wool, Testing the boundaries of the Parrot anti-spoofing defense system, 5th Embedded Security in Cars (ESCAR USA), 2017.

[5] Y. Hamada, M. Inoue, S. Horihata, and A. Kamemura, Intrusion Detection by Density Estimation of Reception Cycle Periods for In-Vehicle Networks: A Proposal, presented at the 14th escar Europe Conference, November 16-17, 2016.

[6] O. Hartkopp, C. Reuber, and R. Schilling, MaCAN - message authenticated CAN, Embedded Security in Cars (escar) 2012, Berlin - Germany, November 2012.

[7] ISO 11898:2015 Road vehicles – Controller area network (CAN), 2015.

[8] 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, In: Security and Privacy (SP), 2010 IEEE Symposium on. IEEE, 2010. p. 447-462.

[9] R. Kameoka, T. Kubota, M. Shiozaki, M. Shirahata, R. Kurachi and T. Fujino, Bus-Off Attack against CAN ECU using Stuff Error injection from Raspberry Pi, Proceedigs of Symposium on Cryptography and Information Security (SCIS), Japan, 2017 (in Japaneses).

[10] C. W. Lin and A. Sangiovanni-Vincentelli, Cybersecurity for the controller area network (CAN) communication protocol, ASE Science Journal, vol.1, No.2, pp.80-92, 2012.

[11] M. Muter and N. Asaj, Entropy-Based Anomaly Detection for In-Vehicle Networks, IEEE Intelligent Vehicle symposium, pp. 1110-1115, 2011.

[12] T. Matsumoto, M. Hata, M. Tanabe, K. Yoshioka and K. Oishi. A method of preventing unauthorized data transmission in controller area network. In IEEE Vehicular Technology Conference (VTC Spring), pp. 1–5. IEEE, 2012.

© 2018 Information Processing Society of Japan

[13] C. Miller and C. Valasek, Adventures in Automotive Networks and Control Units, DEFCON 21, 2013.
[14] C. Miller and C. Valasek, A survey of remote automotive attack surfaces, Black Hat USA, 2014.
[15] C. Miller and C. Valasek, Remote exploitation of an unaltered passenger vehicle, Black Hat USA, 2015.
[16] M. Markovitz and A. Wool, Field Classification, Modeling and Anomaly Detection in Unknown CAN Bus Networks, presented at the 13th escar Europe Conference, November 11–12, 2015.
[17] S. Nie, L. Liu and Y. Du, Free-fall: Hacking TESLA from wireless to CAN bus, Black Hat USA 2016.
[18] D. K. Nilsson, U. E. Larson and E. Jonsson, Efficient In-Vehicle Delayed Data Authentication Based on Compound Message Authentication Codes, Vehicular Technology Conference VTC, 2008.
[19] A. Palanca, E. Evenchick, F. Maggi and S. Zanero, A stealth, selective, link-layer denial-of-service attack against automotive networks, International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Springer, Cham, 2017.
[20] H. M. Song, H. R. Kim and H. K. Kim, Intrusion Detection System Based on the Analysis of time Intervals of CAN Messages for In-Vehicle Network, ICOIN, 2016.
[21] D. Souma, A. Mori, H. Yamamoto and Y. Hata, Counter Attacks for Bus-off Attacks, SAFECOMP 2018 Workshops, LNCS 11094, pp.1-12, 2018.
[22] A. Taylor and N. Japkowicz, Frequency-Based Anomaly Detection for the Automotive CAN Bus, WCICSS, 2015.
[23] A. Wasicek, M. Pese, A. Weimerskirch, Y. Burakova and K. Singh, Context-aware Intrusion Detection in Automotive Control System," presented at the 5th escar USA Conference, USA, June 21–22, 2017.
[24] M. Wolf, A. Weimerskirch and C. Paar, Secure In-Vehicle Communication, Embedded Security in Cars - Securing Current and Future Automotive IT Applications, pp.95-109, 2006.

© 2018 Information Processing Society of Japan