2008－CSEC－43 (3)
2008／12／5

SIPコネクションの第三者による切断問題の検証とその改良方式の提案

田中 真也　木村 成伴　海老原 義彦
筑波大学大学院システム情報工学研究科

高速なネットワークインフラの普及により，一般家庭からインターネットへの常時接続が一般的になってきている．これにともないIPを用いた音声通話サービスであるVoIPへの需要が高まりつつある．VoIPの通話のセッション管理を行うプロトコルとしてSIPがある．SIPで使用されるメッセージは，これを中継するすべてのプロキシサーバでその内容を解析する必要ががあるため，平文で送られるのが一般的である．そのため，SIPメッセージの盗み見やセッションハイジャックなどが起きる可能性がある．本論文では，SIPコネクションが第三者により切断される問題について，AVISPA Toolを用いて検証すると共に，この問題を改良する方式を提案する．

Verification and Improvement of Disconnection Problem of SIP Connection by Attackers

SHINYA TANAKA, SHIGETOMO KIMURA, YOSHIHIKO EBIHARA
Graduate School of Systems and Information Engineering, University of Tsukuba

Nowadays, high-speed network infrastructures are widely spread. Since most home is always connected to the Internet, IP-based telephone service, called VoIP (Voice over IP) is increasingly demanded. SIP (Session Initiation Protocol) is one of the session management protocols of VoIP. In general, SIP messages are transmitted in plaintext, since proxy servers which relay the SIP messages need to analyze the contents of the messages. As a result, an attacker can eavesdrop the SIP messages, and then hijack the SIP connections. This paper uses AVISPA Tool to verify the attacker can disconnect SIP connections, and improves SIP to overcome the problem.

1 はじめに

高速なネットワークインフラの普及により，一般家庭からインターネットへの常時接続が一般的になってきている．これにともないIP (Internet Protocol) を用いた音声通話サービスであるVoIP (Voice over Internet Protocol) への需要が高まりつつある．VoIPでは通常の電話同様に，発信者からの要求に応じて音声通信のセッションを確立したり，切断したりするシグナリングプロトコルが必要であり，その代表的なものとしてH.323[1]やSIP[2]がある．SIPはHTTP (Hyper Text Transfer Protocol) をもとに作成されている．このため，SIPのメッセージはテキストベースであり，そのメッセージ形式は厳密に定められていない．またSIPメッセージは，これを中継するすべてのプロキシサーバでその内容を解析する必要があるため，平文で送られるのが一般的である．そのため，SIPメッセージの盗み見やセッションハイジャックなどが起きる可能性がある．

そこで本論文では，SIPコネクションが第三者により切断される問題について，AVISPA Tool[3]を用いて検証すると共に，この問題を改良する方式を提案する．

本論文での構成は以下の通りである．まず，第2章では，SIPについて概説し，その問題について説明する．第3章では，SIPコネクションの第三者による切断問題の検証を行い，第4章ではこれを解決する方式を提案する．第5章で本論文をまとめると共に，今後の課題を述べる．

2 SIP (Session Initiation Protocol)

SIP (Session Initiation Protocol) は，VoIP のセッション確立，変更，切断などをするシグナリング制御プロトコルである．

SIP では通信の参加者を UAC (User Agent Client) と SIP サーバの二つに分けることが出来る．前者は通話を行うクライアントを指し，SIP 対応電話機や PC 端末がこれに当てはまる．後者はメッセージ解析やアドレスの登録を行うサーバを指し，UAC からのメッセージを解析し，これを通信相手に転送するプロキシサーバや UAC のアドレス管理を行うためのレジストラサーバなど様々なタイプの物が存在する．

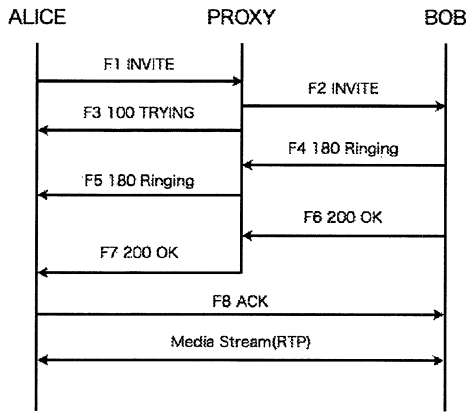

図 2. 1: INVITE メッセージフロー

図 2.1 に，UAC の ALICE がプロキシサーバの PROXY を経由して別の UAC である BOB に対して電話をかけるメッセージフローを示す．まず，ALICE が BOB に対して参加リクエストコマンドである INVITE メッセージを送信する (F1)．PROXY はこのコマンドを BOB に対して配信した後 (F2)，ALICE に対して 100 TRYING メッセージを送信する (F3)．このメッセージはリクエストが処理中であることを表している．PROXY からメッセージを受け取った BOB は，呼び出し中を意味する 180 Ringing メッセージを，PROXY を経由して ALICE へ送信する (F4, F5)．そして，ユーザが参加リクエストを受け付けると，BOB はリクエストがきちんと受理されたことを表す 200 OK メッセージを ALICE へ送信する (F6, F7)．BOB からのメッセージを受信した ALICE は INVITE に対する最終レスポンスである ACK を BOB へと送信し (F8)，ALICE - BOB 間のコネクションを張る事が出来る．

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 INVITE
Contact: <sip:alice@client.atlanta.example.com;transport=tcp>
Content-Type: application/sdp
Content-Length: 151
v=0
o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com
s=- c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

図 2. 2: SIP における INVITE メッセージの詳細

図 2.2 は SIP が通信時に使用するメッセージの詳細についてである．図 2.1 の PROXY にあたるプロキシサーバは図 2.2 の様な形式のメッセージをヘッダフィールドの情報を元に転送をおこなう．しかしそれだけでは無く，プロキシサーバではメッセージ内の解析，例えばリクエストの中継回数を表している Max-Forwards の値を調べて，その値に応じて転送可能かどうかを判断したり，Max-Forwards の値の更新を行ったりしている．そしてこのメッセージ書式は厳密に定められてはおらず，プロキシサーバではメッセージ中継処理のうち，メッセージ解析に対して約 50%が費やされている．そのため暗号化を行うことでさらに負荷が高くなってしまうため，SIP メッセージは平文通信で送られているの．図 2.2 のメッセージの項目に Call-ID という項目がある．この項目は通話のセッションの識別子になっている．そのためこの値を攻撃者に傍受されてしまうと，任意のセッションに対してのメッセージの作成を行えるようになるため，不正なメッセージ送信が行えるようになる．そのため，この事が原因の脆弱性が存在する．以下に脆弱性となりうる問題の例を示す．

- REGISTER リクエストの偽装

  REGISTER リクエストとは，UAC のアド

レスの登録を行うためにアドレスの管理を行っているレジストラサーバにアドレス情報を登録するリクエストメッセージである．攻撃者が不正なメッセージをレジストラサーバに対して送信することで，正しく登録されているアドレスを無効化し，正常に着信を行えなくすることが可能である．

- CANCEL リクエストの偽装

  CANCEL リクエストとは，通話が開始される前にセッションのキャンセルを行い，通信を強制的に終了するリクエストメッセージである．攻撃者が不正な CANCEL リクエストを着信側 UAC(図 2.1 では BOB) に送ることによって，発信側 UAC の意図にかかわらず，セッションの確立を不可にすることが可能である．

- re-invite リクエストの偽装

  図 2.1 で示したように，INVITE リクエストはセッションを確立するために UAC から SIP サーバに送られるメッセージである．re-invite リクエストとは，UAC 同士のセッションが確立されている状態で，メディア情報の変化を行う目的で送られる INVITE リクエストの事を指すが，攻撃者が不正な re-invite メッセージを UAC に送った場合，既存のセッションを乗っ取る事が可能である．

- BYE リクエストの偽装

  BYE リクエストとは，既存の UAC 間セッションを終了する際に UAC から送られるメッセージのことである．攻撃者が不正な BYE メッセージを UAC に送ることによって，既に確立されたセッションを通話者の意図に関係なく切断し，通話を妨害することが可能である．

次章では，これらの攻撃方法の中から BYE リクエストの偽装による攻撃について検証を行い，本問題を解決するための改良方式を提案する．

```
role alice{
        A,B :agent
        K: symmetric_key}
        ...
        SND,RCV:channel(dy)
 }
```

図 3. 1: HLPSL 言語にて記述した Alice の定義

3 問題の検証

本章では，BYE リクエストの不正メッセージによる SIP コネクションの切断問題について焦点を当て，HLPSL (High-Level Protocol Specification Language)[4] にて定義された一般的な BYE リクエストのメッセージフローを，AVISPA Project によって開発された AVISPA Tool (Automated Validation of Internet Security Protocols and Applications)[3] を用いて検証をする．

3.1 AVISPA Tool

AVISPA Tool とは，HLPSL (High-Level Protocol Specification Language) で記述されたメッセージフローの安全性を代数的に解析するアプリケーションである．そこで記述すべき項目として，通信を行うエージェントの定義，エージェントの通信の動作，エージェント同士のセッションの関係，攻撃者の定義，そしてセキュリティの検査を行う項目が挙げられる．

図 3.1 に，エージェント Alice の定義を HLPSL で記述した例を示す．A,B :agent にて通信に使用するエージェント名の定義を行っている．K は対称鍵を表している．SND,RCV:channel(dy) は通信に使用する送信部と受信部を表しており，(dy) は通信内容を攻撃者に盗聴されている事を表している．HLPSL ではこのようにして通信の対象を明示的に記述を行っていく．

3.2 不正リクエストによる SIP コネクションの切断

図 3.2 に，SIP コネクションが正常に切断される際に送られるメッセージフローを示す．この図の開始時において，ALICE と BOB は通話

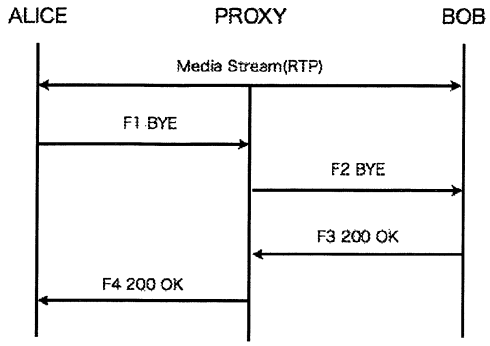

図 3. 2: 正常な BYE メッセージフロー

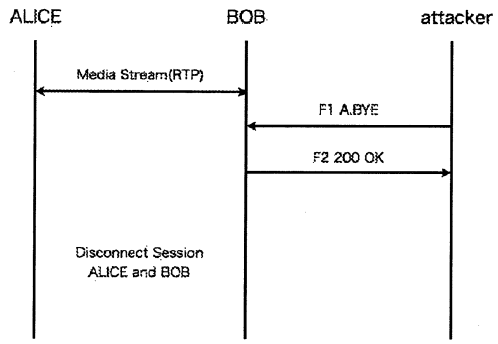

図 3. 3: 不正な BYE メッセージフロー

中の状態である．そして，ALICE は BOB に対してセッションの終了を示す BYE メッセージを送信する (F1)．これを受信した PROXY は，BOB にこのメッセージを中継する (F2)．このメッセージを受信すると，BOB はセッション終了に合意することを表す 200 OK を送信する (F3)．このメッセージも PROXY にて中継され (F4)，これを ALICE が受け取ると通話が終了する．

図 3.3 に，攻撃者が不正に SIP コネクションの切断を行う場合のメッセージフローを示す．図 3.2 の場合とは異なり，攻撃者 (attacker) が不正な BYE メッセージ A.BYE を BOB に対して直接送信をしている．このように，SIP ではプロキシサーバを通さずに，直接メッセージを受け取ることも可能である．そして，その際に攻撃者が送っている BYE メッセージが ALICE になりすましたメッセージであれば，ALICE と BOB の間の SIP コネクションを不正に切断す

```
role alice(
      A,P,B:agent,
      SND,RCV:channel(dy)
      )
 ...
0.    State = 0 /\ RCV(start)
 =|>
      State' :=4 /\ SND(A.BYE)

4.    State = 4/\ RCV(OK')
 =|>
      State' :=7 /\ request
  (B,A,bob_alice_ok,B.OK)
```

図 3. 4: Alice の定義

```
role session(
A,B,P:agent
)

composition
alice(A,P,B,SA,RA)
/\bob(A,P,B,SB,RB)
/\proxy(A,P,B,SP,RP)
}
```

図 3. 5: セッションの定義

ることが可能であると考えられる．

3.3 検証

3.2 節で述べた攻撃が可能であることを証明するため，本節では図 3.2 で示した BYE メッセージフローを HLPSL で記述する．

図 3.4 に，ALICE の定義の一部を示す．A,P,B:agent は ALICE，PROXY，BOB に対応するエージェントを，SND，RCV:channel(dy) は ALICE がメッセージを送受信するチャネル (攻撃者が盗聴可能) を示している．そして，ALICE が可能な状態遷移を 2 つ定義している．最初の定義では，State が 0 であり，RCV が初期状態 (start) であれば，State が 4 の状態に遷移し，そのときに SND から BYE メッセージ (A.BYE) を送ることを表している．次の定義では，State が 4 で

```
role enviroment()
...
  intruder_knowledge
={a,b,p,bye,ok}
  composition
  session(a,p,b)
end role
```

図 3. 6: 通信環境の定義

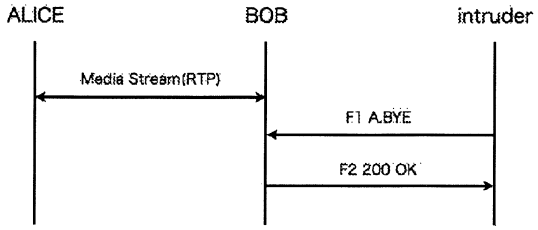

図 3. 7: AVISPA Tool による実験結果

あり，RCV で OK メッセージを受け取ったら，State が 7 の状態に遷移し，この状態における安全性を検証するよう AVISPA Tool に要求 (request) している．同様のことを BOB，PROXY に対しても記述を行っていく．

次に，図 3.5 にセッションの構成部の定義を示す．ここでは，各エージェントが確立しているセッションを記述する．図では，ALICE と BOB と PROXY を合成 (composition) することで，各々のエージェントが互いにセッションを確立している，すなわち互いに通信が可能であることを表している．

図 3.6 に通信環境の定義の一部を表す．ここでは，主に攻撃者が知っている情報 (intruder_knowledge) を定義しており，{a,b,p,bye,ok} で ALICE，BOB，PROXY の存在と，それらがやり取りする BYE メッセージと OK メッセージを知っていること，session(a,p,b) で ALICE，BOB，PROXY 間で送られる通信が盗聴できることを表している．

つまり，攻撃者は ALICE や BOB，PROXY が作成するメッセージと同等のメッセージを送信出来ることを意味している．

3.4 検証結果

図 3.2 を HLPSL 言語を用いて内容を記述した結果，約 100 行で表すことが出来た．これを AVISPA Tool を用いて検証をした結果，図 3.4 の攻撃方法が出力された．これは図 3.3 と同様であり，攻撃者 (AVISPA Tool では INTRUDER と呼んでいる) が ALICE になりすまし，BYE メッセージを BOB に送信している．以上により，不正な BYE メッセージを受け取ることが出来る環境であるならば，第三者が通話中のセッションを切断することが出来る可能性があることが代数的に証明された．

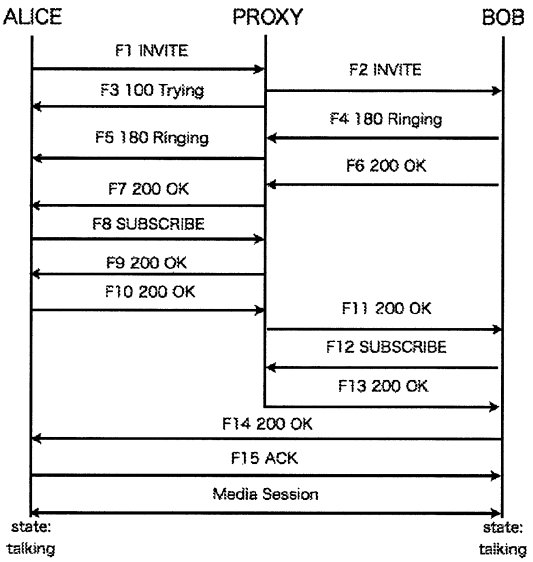

図 4. 1: 提案方式 (1)

4 提案方式

前章では，BYE メッセージに脆弱性がある事が証明された．この問題を解決するために，本章では通話の参加要求のメッセージフロー (図 2.1) と切断のメッセージフロー (図 3.2) の改良方式を提案する．このとき，システムの実装や普及が容易になるように，SIP で現在規定されているメッセージ以外の拡張メッセージは使わないものとする．

図 4.1 に，セッション開始時のメッセージフローを示す．まず，図 2.1 で行った通話の参加要求 (F1–F7) を行った後，ALICE と BOB は PROXY にたいして SUBSCRIBE 要求をしてい

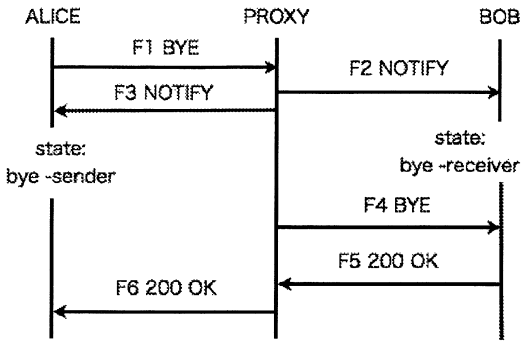

図 4. 2: 提案方式 (2)

る (F8–F14). これにより，PROXY は特定のイベント (この場合は BYE メッセージの受信) が起きると，NOTIFY リクエストを ALICE と BOB に通知するようになる．以上により，セッションが確立されると，ALICE と BOB は自身の state を talking に変更する．ここで state とは通話の状態を表すもので，主に SIP で使用しているインスタントメッセージサービスにて使われている．state を talking (通話中) としたため，ALICE も BOB も全ての BYE メッセージを拒絶するようになった．これにより，攻撃者からの不正な BYE メッセージの返信が防がれる．

次に，図 4.2 にセッション終了時のメッセージフローを示す．ALICE が BYE メッセージを PROXY に送信すると (F1)，PROXY は BOB と ALICE に対して NOTIFY 通知を行う (F2, F3). この通知を受け取った BOB は state を bye-receiver に変更し，BYE メッセージの受信を許可する．ALICE は state を bye-sender に変更し，BYE メッセージの応答を受け付ける．その後，図 3.2 と同様なフローが行われ (F4–F6)，セッションの切断が完了する．

最後に，図 4.3 に攻撃者が不正切断を試みようとした場合のメッセージフローを示す．BOB は state が talking であるため，話し中を表す 486 Busy here を返信し (F2)，不正切断が防止される．

5 まとめ

今回 AVISPA Tool を用いて，SIP コネクションの第三者による切断が可能であることを代数

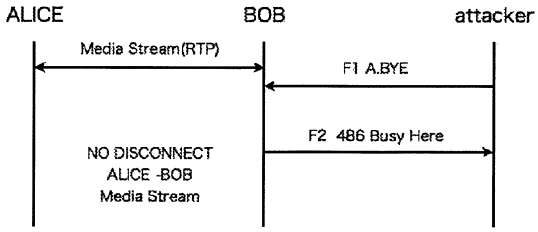

図 4. 3: 攻撃者が不正メッセージを送信した時

的に証明するとともに，この問題に対する改良方式の提案を行った．今後の課題として，提案方式を HLPSL で記述し，AVISPA Tool にて検証をすることや，また，本提案方式を本研究室で開発した Light Weight SIP[6] に適用することも検討している．

参考文献

[1] ITU-T Recommendation H.323, "Packet Based Multimedia Communications Systems," Feb. 1998.

[2] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R .Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," RFC 3261, Jun. 2002.

[3] The AVISPA Project,
http://avispa-project.org.

[4] The High Level Protocol Specification Language,
http://avispa-project.org/delivs/2.1/d2-1.pdf.

[5] AVISPA v1.1 User Manual,
http://avispa-project.org/package/user-manual.pdf.

[6] F. Satoshi, K. Shigetomo and E. Yoshihiko, "A Proposal of Lightweight SIP for VoIP," Vol.105, No.408, pp. 37–42, IN2005-106, Nov. 2005.