SASの同期問題に関する研究

1225122　高橋 錬　【 セキュリティシステム研究室 】

A study on SAS Asynchronous problems

1225122　Takahashi Ren　【 Security Systems Lab. 】

1　はじめに

IoT の普及に伴い，IoT デバイスへの攻撃が増加している [1]．また，IoT デバイスの中には処理能力などの制限が厳しいものも存在する．そこでこれらのデバイスには軽量なセキュリティ技術が求められる．軽量なセキュリティ技術として SAS が提案されている．

SAS は共通鍵暗号方式をベースとしており，相互認証・鍵配送・暗号通信を実現している． SAS を導入した場合，デバイス間の通信経路が何らかの原因で遮断されることで認証情報にズレが生じる可能性がある．これを同期問題という．この同期問題によって可用性が損失することはシステム全体に大きな影響を与える．また，この同期問題を意図的に引き起こすことで攻撃にもなり得るため，解決する必要がある．同期問題解決手法として中原らの方式と藤田の方式が提案されている．これらの方式では可用性を満たす一方でなりすましによる安全性への課題が存在する．

本稿では既存方式の課題を解決し，可用性と安全性を満たす方式を提案する．また，同期問題対策による拡張が処理時間に与える影響についても調査し，拡張による処理への影響が限りなく小さいことを示す．

2　SAS

SAS はワンタイムパスワード認証方式であり，その他の共通鍵暗号方式と比較して一方向性関数の適用回数が少ないことが特徴である [2]．SAS は通信路上を流れる認証情報が毎回変化することからリプレイ攻撃やなりすましへの耐性がある．SAS にはいくつかのバージョンかがあり，相互認証の実現や IoT デバイス向けに初期の SAS よりもさらに一方向性関数の適用回数を削減した方式が提案されている．SAS は初回登録フェーズと認証フェーズで構成されている．初回登録フェーズは安全な通信路を用いて行われる．

2.1　同期問題

SAS-2 を導入する際には同期問題への対策を取る必要がある．同期問題とは何らかの影響でパケットが相手側の端末に到達しなかった場合に端末間の認証情報にズレが生じることで次回以降の認証ができなくなる問題のことである．これは意図的に通信を遮断し同期問題を引き起こす攻撃 (同期ズレ攻撃) につながるためこの問題への対策は重要となる．

以下で SAS の認証フェーズを図 1，同期問題の概要図を図 2 に示す．

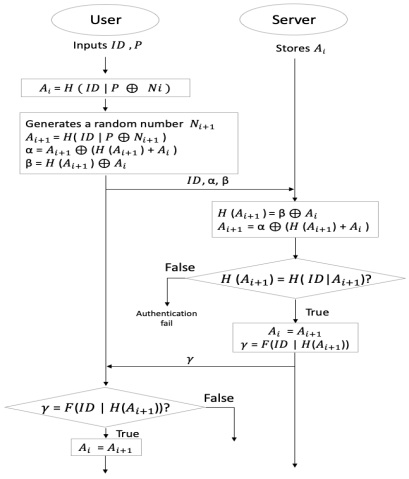

図 1　SAS-2 の i 回目認証フェーズ

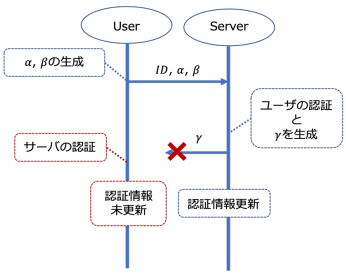

図 2　同期問題発生時の認証フェーズ

3　既存方式

既存方式として中原らの方式と藤田の方式がある．中原らの方式は前回利用した認証情報をバックアップすることで拡張による影響を最小限にし，同期問題を解決している [3]．しかし，攻撃者によるリプレイ攻撃への対策が課題となっている．

藤田の方式はチャレンジ&レスポンス方式を採用することでリプレイ攻撃に強く，同意問題を解決する方式を実現している [4]．しかしその一方で，藤田の方式では

サーバ側になりすますことが可能となる．サーバになりすますことでユーザ側のデバイスに不正にアクセスされる恐れがあるため対策が必要である．

4　提案方式

本稿では藤田の方式の課題を解決し，安全性と可用性を満たす方式を提案する．提案方式では藤田の方式で用いられているチャレンジの値をリストに記録し，サーバ側で認証後に生成される γ の値にチャレンジの値を組み込むことで課題を解決した．以下に提案方式の認証フェーズを図 3 として示す．

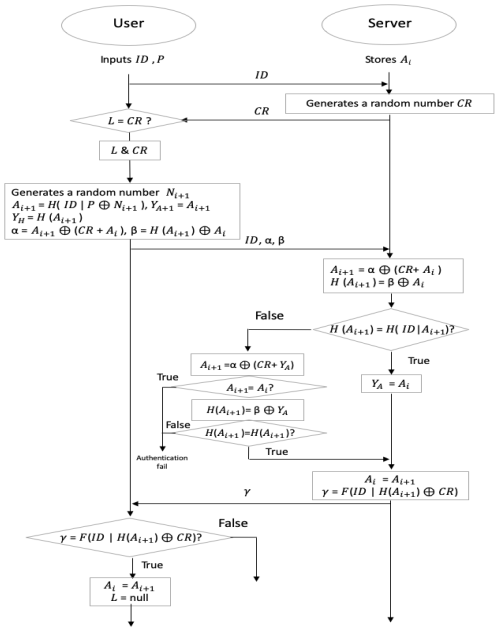

図 3　提案方式の i 回目認証フェーズ

5　実験結果

本実験は同期問題対策を行った場合の処理時間への影響を調査することが目的である．比較対象は SAS の各バージョン対して同期問題対策の有無で調査した．内部処理時間は通信を除いた時間であり，総処理時間は通信を含めた時間である．測定値は 10 回測定した平均値を示している．使用言語は Python である．

実験環境を表 1 に示す．また，実験結果を図 4 と図 5 に示す．

	ユーザ側	サーバ側
端末	Raspberry Pi ZERO WH	MacBook Pro
メモリ	512MB	8 GB
OS	Raspbian	Mac OSX
CPU	160MHz	2.7GHz

表 1　ユーザ側環境

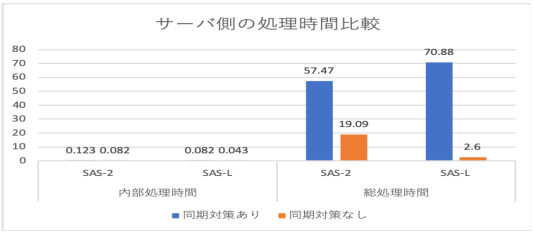

図 4　同期問題発生時の認証フェーズ

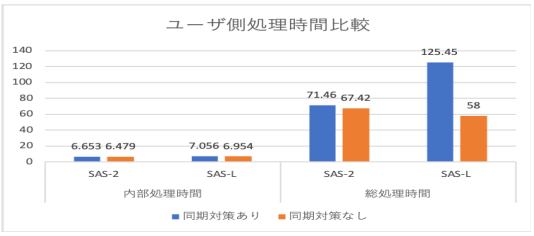

図 5　同期問題発生時の認証フェーズ

実験の結果から提案方式は通信回数に伴う通信処理時間の影響を受けることから即時性が求められる環境には適さない．しかし，内部処理の処理負荷は数ミリ秒と小さいため内部処理負荷を下げることを目的にサーバに接続する端末数が多い場合に有効であると言える．

6　まとめ

本稿では，IoT デバイスへの適用を目的に軽量なセキュリティ技術である SAS の可用性と安全性を確立するための拡張を行った．今後の課題として一般的に普及している TLS 等のセキュリティ技術と比較して有用性を示すと共に適用範囲の検討を行うことがあげられる．

参考文献

[1] “第一部　第一節　世界と日本の ICT 市場の動向,” 情報通信白書平成 30 年版:世界と日本の ICT，総務省,p.7，https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h30/pdf/n1100000.pdf，(参照 2020-1-15)

[2] T. Tsuji and A. Shimizu, “A One-Time Password Authentication Method for Low Spec Machines and on Internet Protocols,” IEICE Trans. COMMUN, vol. E87-B, no.6, pp. 1594—1600, 2004.

[3] 中原知也, 辻貴介, 清水明宏, “SAS-2 認証方式の同期問題に関する検討,” 電子情報通信学会技術研究報告, OIS, vol.104, no.714, pp.83-87, 2005.

[4] 藤田 寛泰，“SAS-2 の同期問題対策時における なりすまし防止に関する研究，” 高知工科大学学士学位論文，2017