GASP : a Generic Approach to Secure network Protocols

1st Olivier Levillain
Samovar, Télécom SudParis, Institut Polytechnique de Paris
olivier.levillain@telecom-sudparis.eu

2nd Aina Toky Rasoamanana
Valeo
aina.rasoamanana@valeo.com

***Résumé*—Dans notre monde toujours plus connecté, nous reposons fortement sur les protocoles réseau pour communiquer les uns avec les autres. Pour assurer la confidentialité et l'intégrité des données échangées, il est donc nécessaire d'évaluer et d'améliorer la sécurité de ces protocoles et de leur implémentation. Cette observation concerne évidemment les protocoles dits de sécurité tels que TLS ou SSH, mais il est également important de s'intéresser aux protocoles applicatifs complexes de plus haut niveau, comme HTTP/2, et aux protocoles de plus bas niveau comme DNS ou BGP.**

Les failles de sécurité sont en effet omniprésentes dans nos protocoles réseau, que ce soit au niveau de la spécification ou à cause d'erreurs d'implémentation : spécifications incomplètes, problèmes de corruption mémoire, erreurs logiques, raccourcis dans les machines à état, attaques cryptographiques, etc.

Le projet GASP, financé comme un projet JCJC par l'ANR entre 2019 et 2024, a permis d'explorer plusieurs pistes liées à la sécurisation des implémentations des protocoles réseau en utilisant diverses méthodes : étude des générateurs de *parsers*, implémentation de piles modulaires, analyse des machines à états d'implémentations concrètes pour TLS et SSH.

Ces travaux s'inscrivent dans une activité plus générale de l'équipe « Sécurité et Confiance Numérique » du laboratoire Samovar à Télécom SudParis.

***Index Terms*—Sécurité des protocoles réseau, inférence de machines à états.**

I. Introduction

Les années 2010 ont été particulièrement intéressante pour l'étude du protocole TLS, puisque de nombreuses vulnérabilités ont été mises au jour durant cette décennie. En particulier, de nombreuses failles liées aux machines à états ont été publiées en 2014 et 2015. On peut citer les vulnérabilités suivantes, qui ont été particulièrem ent médiatisées :

- CVE-2014-0224 (*EarlyCCS* [Kik14]), qui permettait de rendre la clé de session prédictible entre un client et un serveur tous les deux vulnérables ;
- CVE-2015-0204 (FREAK, *Factoring RSA Export Keys* [BBD+15]), qui forçait l'utilisation de vieilles suites RSA EXPORT (i.e. avec des clés RSA de taille réduite) entre un client vulnérable et un serveur acceptant les suites EXPORT ;
- CVE-2014-6321 (Winshock), un débordement de tampon menant à une exécution de code arbitraire ; même si cette vulnérabilité n'est pas en soi un problème de machine à états, un laxisme dans la machine à états rendait cette faille dans l'authentificat client par certificat accessible sur des serveurs IIS n'ayant pas sollicité de certificat client.

En 2018, une vulnérabilité intéressante a également été publiée sur `libssh`, une implémentation du protocole SSH. Durant la phase d'authentification de l'utilisateur [1], un serveur vulnérable acceptait le message `UserAuthSuccess` (un message qui n'est normalement jamais envoyé par le client) en lieu et place d'une requête d'authentification (le message `UserAuthRequest`). Là encore, cette faille démontre l'existence de problèmes critiques dans les machines à états des implémentations des protocoles réseau.

De ce constat est né l'idée d'étudier de manière systématique les implémentations de différents protocoles réseau comme TLS et SSH, ce qui a donné le projet GASP, dont les données techniques sont données dans le tableau suivant.

Nom	GASP (a Generic Approach to Secure network Protocols
Porteur	Olivier Levillain (Télécom SudParis)
Consortium	Télécom SudParis [2]
Instrument	ANR JCJC (Jeune Chercheur Jeune Chercheuse)
TRL	3-5

GASP a été l'occasion de recruter Aina Toky Rasoamanana (co-auteur de cette contribution) en thèse, entre 2019 et 2023.

Dans le reste de ce document, nous allons insister sur les travaux que nous avons menés dans le cadre de GASP sur l'inférence des machines à états TLS et SSH.

II. L'*Active Automata Learning*

Nos travaux s'inscrivent dans le domaine de l'*Active Automata Learning*, une approche qui consiste à interagir avec la cible de notre étude (souvent appelée SUL, pour *System Under Learning*), afin d'en déduire le fonctionnement interne sous la forme d'une machine à états. La figure II décrit le fonctionnement de cette approche.

L'idée principale est de modéliser le comportement de la cible à l'aide de messages abstraits, de soumettre des

Ce projet a été financé par l'ANR sous le numéro ANR-19-CE39-0001.

1. La couche *UserAuthentication* est le deuxième étage de la fusée dans une connexion SSH. Elle vient après la couche *Transport* qui met en place un canal chiffré et avant la couche `Connection` qui permet l'établissement de canaus applicatifs.

2. Au sens strict, Télécom SudParis était le seul partenaire puisque GASP était un projet ANR JCJC, mais le projet a été l'occasion de discussions intéressantes avec différentes équipes, par exemple à l'ANSSI et au Loria.

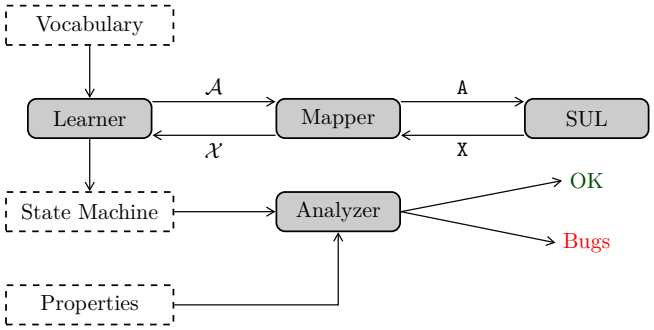

FIGURE 1. Description des différents composants de notre pipeline.

séquences de messages à la cible, et d'inférer sa machine à états interne. Bien entendu, pour communiquer avec la cible, il est nécessaire de traduire les messages abstraits en messages concrets (et inversement), à l'aide d'un composant appelé le *mapper*. Enfin, une fois les machines à états obtenus, il faut les analyser pour mettre en évidence les écarts à la spécification et les éventuelles failles de sécurité.

III. RÉSULTATS

Lors de la présentation à RESSI, nous prévoyons, après avoir brièvement rappelé les enjeux du projet et le cadre de l'*Active Autoamta Learning*, de présenter quelques résultats obtenus durant nos travaux :

- l'inférence systématique et rigoureuse de nombreuses piles TLS [RLD22], qui a donné les résultats suivants :
  - l'amélioration des performances grâce à des optimisations exploitant le déterminisme des machines inférées (division par 25 du temps d'inférence dans le meilleur cas),
  - la mise au jour de nouvelles vulnérabilités (e.g. CVE-2021-3336, illustré à la figure III),
  - la démonstration que les différences entre machines à états pouvait permettre de prendre une empreinte et de réidentifier (*fingerprinting*) les piles TLS à partir de leur comportement ;
- l'inférence de machines à états SSH avec une plus grande expressivité que les travaux précédents (ces travaux seront prochainement soumis) :
  - l'écriture de plusieurs *mappers* SSH permettant d'explorer en détails certains comportements des implémentations
  - la reproduction automatique de vulnérabilités connues ;
  - la mise au jour d'une vulnérabilité critique dans le client wolfSSH (CVE-2025-14942).

Le temps de présentation nous permettra normalement de proposer une démonstration de la vulnérabilité CVE-2025-14942, et des différents scénarios d'attaque possibles.

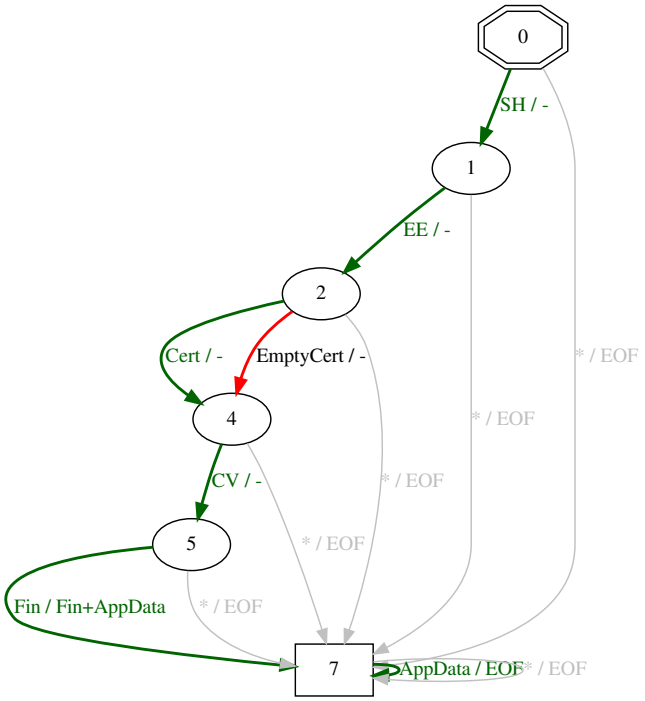

FIGURE 2. CVE-2021-3336 : en envoyant un message `Certificate` vide au lieu du certificat du serveur, un attaquant peut contourner l'authentification auprès du serveur ; il lui suffit d'envoyer un message `CertificateVerify` (`CV`) quelconque pour terminer la négociation.

IV. ET APRÈS ?

Comme indiqué plus haut, le projet GASP s'inscrit dans un axe de recherche à Télécom SudParis, qui a fait l'objet d'autres projets et d'autres résultats :

- le projet CERES (financé par le CIEDS) a financé la thèse d'Arthur Tran Van ; ce dernier a travaillé sur une approche systématique et rigoureuse pour l'analyse de machines à états, le *Mealy Verifier* [VLD24], ainsi que sur des techniques d'apprentissage adaptatives qui font l'objet d'une soumission en cours ;
- le projet GINS (financé par Carnot Télécom) a financé le postdoc de Yohan Pipereau, qui a travaillé sur des technique d'instrumentation en boîte grise, pour améliorer la précision de l'inférence et mettre au jour des états profons inaccessibles aux techniques classiques (soumission en cours).

Des extensions sont en cours de montage, pour améliorer les performances (suite des travaux sur l'apprentissage adaptatif et le travail en boîte grise, la captation des appels système pour réduire les temps d'attente) et pour étendre les cas d'usage (nous voulons utiliser notre joli marteau sur tous les clous que nous pouvons trouver !).

RÉFÉRENCES

[BBD+15] Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, Alfredo Pironti, Pierre-Yves Strub, and Jean Karim Zinzindohoue. A messy state of the union : Taming the composite state machines of TLS. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA*, pages 535–552, May 2015.

[Kik14] Masashi Kikuchi. How I discovered CCS Injection Vulnerability (CVE-2014-0224). http://ccsinjection.lepidum.co.jp/blog/2014-06-05/CCS-Injection-en/index.html, June 2014.

[RLD22] Aina Toky Rasoamanana, Olivier Levillain, and Hervé Debar. Towards a Systematic and Automatic Use of State Machine Inference to Uncover Security Flaws and Fingerprint TLS Stacks. In *27th European Symposium on Research in Computer Security, ESORICS 2022, Copenhagen, Denmark*, pages 637–657, September 2022.

[VLD24] Arthur Tran Van, Olivier Levillain, and Hervé Debar. Mealy Verifier : An Automated, Exhaustive, and Explainable Methodology for Analyzing State Machines in Protocol Implementations. In *Proceedings of the 19th International Conference on Availability, Reliability and Security, ARES 2024, Vienna, Austria*, pages 1–10, July 2024.