CS 298 Proposal : EAP-SIM Vulnerabilities and Improvements

Guided by Dr. Thomas Austin

By :
Akshay Baheti
San José State University

July 6, 2015

Abstract

Today there is an the exponential increase in the number 3G/4G enabled devices. A cell tower can support only a limited number of 3G/4G connection at a given time. This limitation of mobile technology leads to degraded service quality for customers at social gathering. WiFi is a solution to the problem as it can support large number of clients compared to Cellular data. Using WiFi to serve cellular clients is known as 3G Wifi Offloading[1]. Authentication over Wifi is a challenge though. This document describes the Extensible Authentication Protocol and several of its best-known security issues. This protocol is widely used for authentication over Wireless Networks. The document also introduces the basic functionality of EAP for Subscriber Identity Module(SIM) and discusses several of it's vulnerabilities. It later discusses possible improvements to avoid attacks of EAP-SIM.

1 Project Deliverables

The list of project deliverables includes the following

- Detailed procedure for carrying out the attack
- wap-supplicant code with changes to showing high probability of the attack
- Server-side code with the defence of the attack
- Testcases and results showing improved protocol

2 Challenging aspects of Project

The main challenge involved is implementing an attack with a high success rate. The attack on EAP-SIM requires understanding the protocol and open source implementation of the same. Driving the defence requires a understanding both the client and server side implementation of the protocol. There are various open source implementations of the EAP protocol which vary significantly. Hence the attack implementation is a major challenge.

Following the implementation find a suitable defence to the attack trying out the defence in the opensource is the next milestone for the project. Building a defence requires a through understanding of the protocol and the open source tool. Once a defence is ready ensuring that the defence does not significantly hurt the performance on the protocol remains a issue.

3 Schedule

Table 1: Timeline

Week 1: May 4th - May 8th	Understand the EAPSIM protocol
Week 2-3: May 11th - May 22nd	Try to set up a basic freeradius setup for EAP SIM
Week 4: May 25th - May 29th	Configure router for EAPSIM to work with WiFi and free radius
Week 5: June 1st - June 5th	Configure opensource tool to run with EAP-SIM
Week 6-7: June 8th - June 19th	Prepare for EAP-SIM DoS attack setup
Week 8-9: June 22nd - July 3rd	Try out the EAP-SIM attack
Week 10-11: July 6th - July 17th	Find possible improvements/changes in protocol to avoid the attack
Week 12-13: July 20th - July 31st	Find code point to try out the improvement
Week 14-15: Aug 1st - Aug 15th	Code the improvements in Hostapd
Week 16-17: Aug 17th - Aug 28th	Code the improvements in wpasuplicant
Week 18-19: Sept 1st - Sept 11th	Write test cases to test performance
Week 20-21: Sept 14th - Sept 25th	Carry out test to show performance of improvements
Week 22-23: Sept 28th - Oct 19th	Study other possible attacks on EAP
Week 25: Oct 19th - Oct 23rd	Study other possible improvements on EAP
Week 26-27: Oct 26th - Nov 6th	Write report
Week 28-29: Nov 9th - Nov 20th	Prepare for defence

References

[1] Bertrand Meyer. Applying design by contract. *Computer*, 25(10):40–51, 1992.

[2] Bertrand Meyer. *Eiffel: the language*. Prentice-Hall, Inc., 1992.

[3] Richard Mitchell, Jim McKim, and Bertrand Meyer. *Design by contract, by example*. Addison Wesley Longman Publishing Co., Inc., 2001.

[4] Jeffrey E Payne, Michael A Schatz, and Matthew N Schmid. Implementing assertions for java. *Dr. Dobb's Journal*, 23(1):40–44, 1998.

[5] Richard S Wiener. *Software development using Eiffel*. Prentice-Hall, 1995.