Elliptic Curves in Transport Layer Security (TLS)

Project Abstract

Balakrishnan Vasudevan

balakrishnanvasudevan@umail.ucsb.edu

November 1, 2015

ABSTRACT

By the end of the year 2016, approximately 2.94 billion people and more than 25 billion devices will have accessed 1.1 zetabytes of data that includes anything from movies on Netflix or papers on arXiv to grocery shopping and cash transfers on the internet[1]. Securing these transactions is of utmost importance. In February 2015, it was discovered that the insurance provider Anthem was the target of an attack that resulted in more than 80 million records of Social Security numbers, email and physical addresses being stolen. Transport Layer Security (TLS) is a mechanism used to provide privacy and data security between two communicating applications[2]. All major web browsers provide support for TLS to secure communications between them and the web servers.

TLS ensures that the communication between the two applications is private using symmetric cryptography. Public Key Cryptography is optionally used to authenticate the identity of communicating devices. TLS also provides Message authentication to ensure reliable communication between the applications.

This tutorial explains the TLS algorithm for key exchange, ciphering and message authentication. The various Elliptic Curve cryptographic functions being used in the current version TLS 1.2 and the draft version of TLS 1.3 are explained. It also discusses the strengths and vulnerabilities of algorithms like Elliptic Curve Diffie Hellman, Elliptic Curve Ephemeral Diffie Hellman and Elliptic Curve Digital Signature[3].

REFERENCES

[1] Cisco Systems Inc. *The Zettabyte Era-Trends and Analysis - Cisco Visual Networking index*.

[2] T. Dierks, E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246, IETF, August 2008.

[3] S. Blake-Wilson et al. *Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)*. RFC 4492, IETF, May 2006.