Mathematical Models for Encrypted Internet Communication

Ryan Jensen

November 9, 2014

Abstract

Any encryption methods employed for public use on the Internet must maintain their integrity while respecting the following three axioms: (i) all communications can be intercepted and modified in transit; (ii) all algorithm's details must be publicly available to users and attackers alike; and (iii) no communication may bypass the network to trade information secretly at any point in the correspondence. Although these axioms rule out all symmetric encryption systems as potential candidates, advanced number theory and asymmetric encryption systems can provide the initial steps to start an encrypted session and allow symmetric encryption to take over at that point. More specifically we will be exploring the RSA Cryptosystem and its implications on a private Internet, digital authenticity, and using mathematical models to show how and why it works.

1 Introduction

The goal of this paper is to ease the reader into the complex subject of encrypted Internet communication. First we will attempt to shed some light on the challenges of achieving privacy on a public network and look into a particular solution to the problem at hand–The RSA Cryptosystem. The context of this paper is the modern day Internet, which we will define loosely as a network of interconnected computers which are both utilizing the network (consuming) and sustaining the network (producing). In order to communicate with any other person or computer on the network, our traffic must necessarily be routed from one computer to another until it is routed to the correct destination; think of the network as a graph, the computers as the vertices and the connections between particular computers as the edges. Although most of the computers a message is relayed between are Internet service providers, commercial companies, or benign users, the troubling fact is that any one of them could be a malicious attacker looking to access and exploit the information that passed through their computer on its way to the intended destination. We will refer to this type of an attack as a *man-in-the-middle attack*.

Encoding schemes are a necessary step to encryption since most encryption algorithms rely on math functions that operate only on numbers. Typically a message is made up of a string of characters or *glyphs* so we will need a function that maps a glyph to a particular integer. We would call such a function an *encoding* of a set of characters; to map from an integer back to a glyph is a *decoding* function. A simple example of such an encoding would be to let $A = 01, B = 02, ..., Z = 26$; now we can get from characters to integers interchangeably. Common encoding schemes used on the Internet are *UTC-8* and *UTC-16* which encode 2^8 and 2^{16} different characters respectively.

Historically encryption is introduced with a mention of the *Caesar Cipher* or the *shift cipher*; This is a basic form of encryption where you take each encoded character and add a specified amount to the value and wrap values too high back to the beginning (modulus); we could also conceptualize this operation as a character shift. We will define our *secret* or our *key* to be the piece of information required to encrypt or decrypt messages. For the shift cipher the key would be the number of characters we shifted. The shift cipher is an example of a *symmetric* encryption system, which means it uses the same key for encryption and decryption on both *endpoints* (the sender and receiver of the encrypted communication). By contrast, *asymmetric* encryption systems have a key or secret for each endpoint; each key has two components: one for encryption and one for decryption. In total, a encrypted conversation between two endpoints, denoted A and B, using an asymmetric system would have four pieces: encryption key from A to B, encryption key from B to A, decryption key for A, and a decryption key for B.

The Internet