Christopher LaJon Morgan

Oct 23, 2013

Homework 9 CS 465

**Weak RSA Moduli bug**

The error discovered was a lack of true randomness used in generating RSA primes. They found that by factoring a number of already exiting RSA keys (specifically their n values) they were able to discover one of the secret primes used during key generation. This was possible because they found that across multiple implementations the generation of RSA’s prime numbers was not completely random. This lead to the easy regeneration of the RSA components though means such as GCD on n’s of multiple RSA keys, this gave them one of the two primes, and thus the crack of these RSA keys.

In practice, however, this error was not threatening in the least. This was because the vast majority (maybe all but one) of the comprisable keys were found in already vulnerable certificates. In other words, the comprisable keys were found in certificates that were either expired or not signed by a valid CA. The article even made the claim that “this attack almost certainly affects not a single production website.”

Even though the immediate treat of this vulnerability is thought to be minimal, it still brought up some valuable lessons. One of the lessons that I learned while reading this article is how complex the idea of security is. In other words, a powerful and correct algorithm, with a correct implementation, can be broken by simple placing it in mass aggregate. The flaw here was simply an implementation when spread across multiple devices. Therefore, the new idea I gained was that security not only has to be thought of as a case by case bases but also as a collective whole. Is the implementation secure? Is the collective use of the implementation also secure?