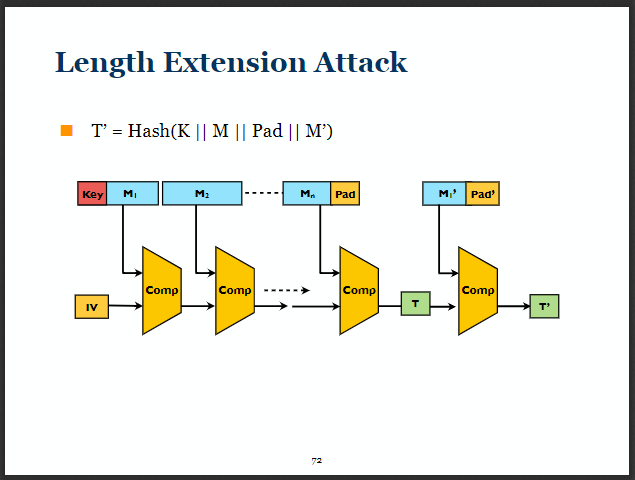
SM3 长度扩展攻击实验报告

SM3是由我国国家密码局发布的密码杂凑（HASH）算法，主要用于数字签名及验证、消息认证码生成及验证、随机数生成等。SM3是散列函数的一种，属于单向函数。通常称其输入数据为 消息；其输出数据为 消息摘要 或 摘要。

加密过程如下：

首先将消息进行分块，每块的消息长度相等，对于最后一个不足规定长度的消息进行补充，并在最后填上实际消息长度的相关信息。将一个初始值IV以及第一个消息块做为输入，放入一个复杂压缩函数进行压缩，将其输出结果做为新的初始值与第二个消息块一起做为输入放入压缩函数进行压缩，再将输出结果做为新的初始值与下一个消息块做为输入进行压缩……如此循环往复直至所有消息块都压缩完毕所得到的输出即为SM3算法的加密结果。

长度扩展攻击，是基于SM3对消息进行分组并多轮压缩的原理，在原有消息后附加恶意消息或是攻击、干扰内容。并且只需要取得原消息的输出作为新的初始值与攻击内容一起作为输入进行新的多轮的压缩，直至附加消息都加密完成。

如上图所示，原消息加密完成后所得到结果为T，我们继续使用T做为初始值，对附加消息M‘（已经补充成完整消息块）进行加密，最后得出一个合法的T’。

长度扩展攻击实践：

实验中SM3加密源码参考自：<https://github.com/NEWPLAN/SMx>

实验目标：原消息：”Henry”

附加消息：”hhhh”

最终目标：最终得到一条合法加密的如下消息：”Henry\00x80\0……hhhh”

程序结构：SM3.cpp、SM3.h实现SM3加密过程。

SM3\_Attack.cpp、SM3\_Attack.h对附加消息进行填充并且对附加消息进行加密。

Main.cpp中进行操作：初始化IV值存入SM3\_Context中，对”Henry”进行SM3加密。此时的SM\_Context中留存有原消息加密的结果，此时不要对它进行初始化，直接对附加消息进行填充，然后使用SM3\_Context中的值进行新的加密，即可达到消息扩种攻击的效果。验证攻击效果，生成验证用的消息”Henry\00x80\0……hhhh”，对SM3\_Context初始化并进行SM3加密。将两个Hash值进行比较，如果相等则攻击成功。

实验截图如下：

