



哈爾濱工業大學(深圳)

HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN

# Hash 长度扩展攻击





## 实验目的

---

- 掌握 Hash 函数的计算原理和算法特点
- 理解 Hash 长度扩展攻击的工作原理
- 了解抵御 Hash 长度攻击的方法



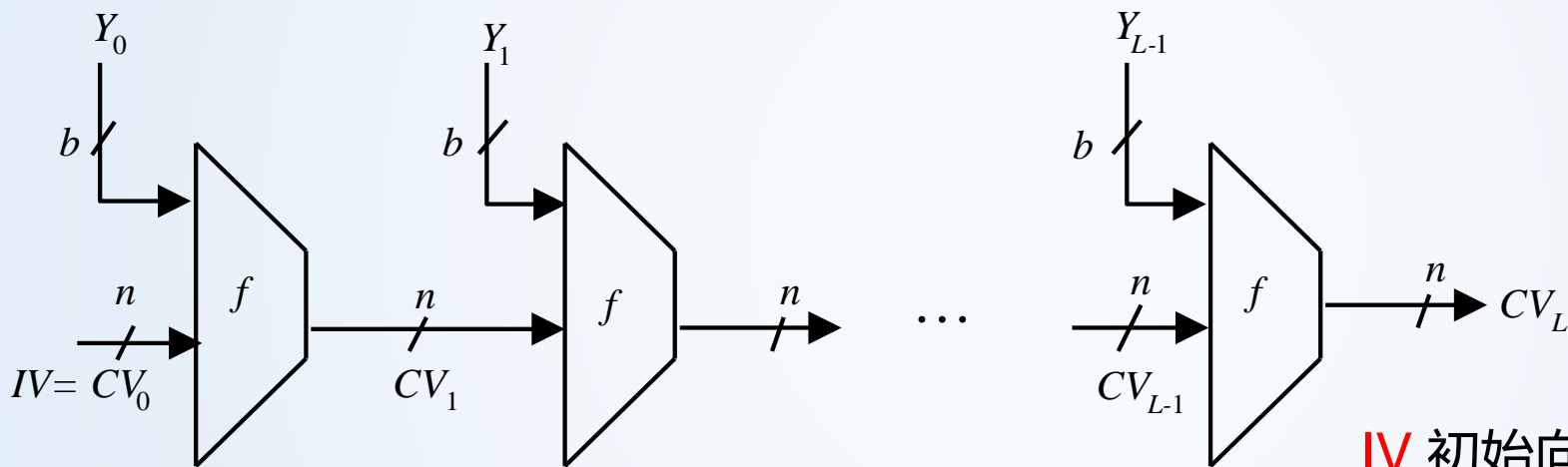
## 实验原理 ---基于哈希函数的MAC

将密钥和消息一起作为哈希函数的输入求哈希值，本次实验是针对秘密前缀Hash算法产生的MAC进行攻击。

- $T = \text{MAC}_k(M) = h(k||M)$ , 称为秘密前缀(secret prefix) MAC
- 设消息  $M$  为一个分组序列(sequence of blocks), 即  $M = (x_1, \dots, x_n)$
- 则  $T = \text{MAC}_k(M) = h(k||x_1, \dots, x_n)$
- 此种构造方式的问题在于, 一致消息  $M$  的 MAC 值和  $k$  的长度, 消息  $M' = (x_1, \dots, x_n, x_{n+1})$  (其中  $x_{n+1}$  为任意的敌手设定的分组) 的 MAC 值可以直接由  $T$  计算得出而不需要知道密钥  $k$  的值。



**迭代型哈希函数的一般结构-MD结构，这种构造方式是每个消息块都会和一个输入向量做运算，第一个输入向量是初始化的，后面每个输入向量都是前面一个消息块的输出结果。**



### b 输入分组长度

$$CV_0 = IV = n$$

$$CV_i = f(CV_{i-1}, Y_{i-1}) \quad 1 \leq i \leq L$$

$$H(M) = CV_L$$



## 实验原理—SHA256算法过程

## ➤ 预处理

- ◆ 对消息填充
- ◆ 初始化缓冲区，缓冲区为8个寄存器共256位

➤ **压缩过程**，循环处理L个消息分组，每个分组处理分为如下两步

- ◆ 首先将消息 $M_i$ 分为16个32位的字，并将其扩展为64个32位的字 $W_0 - W_{63}$ ；
- ◆ 然后将上一轮的输出 $H^{i-1}$ 与产生行64轮次的循环，最后输出256位 $H^i$ 为本次迭代的输出。

## ► 输出结果

- ◆ L个分组都被处理完后，最后一个H<sub>SHA256</sub>的输出即为产生的消息摘要

## ➤ 预处理—消息填充

因为SHA256是512bit一个分组，所以填充后消息长度必须为512的整数倍

◆ 首先在末尾处附上64比特消息长度

◆ 然后在消息原文后面填充，第一位为1，其余为0，至少需要填充1位，如果原始消息长度加上64bit消息长度刚好是512的整数倍，那么需要填充512位。

◆ 填充后的消息必须恰好为512的整数倍

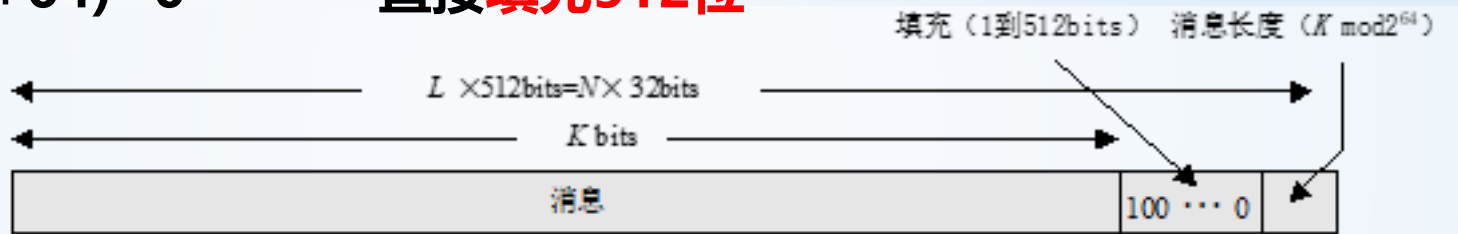
## ➤ 举例说明，计算需要填充的长度

◆ 700位 ->  $1024 - (700 + 64) = 260$

填充260位

◆ 960位 ->  $1024 - (960 + 64) = 0$

直接填充512位



**Tips: 消息填充是必须的，在任何情况下都需要消息填充**











## 实验内容

本次实验来自[https://seedsecuritylabs.org/Labs\\_20.04/Crypto/Crypto\\_Hash\\_Length\\_Ext/](https://seedsecuritylabs.org/Labs_20.04/Crypto/Crypto_Hash_Length_Ext/)  
共需要完成如下4个小任务。通过这4个任务完成对一个服务器网站的 Hash 扩展攻击，并对 Hash 扩展攻击进行防御验证。

- 任务1：发送请求来列出所有文件
- 任务2：创建Padding
- 任务3：长度扩展攻击
- 任务4：使用 HMAC 来抵御长度扩展攻击



# 实验要求

- 提交内容

- ① 实验结果截图

- 截止时间

一周内提交至HITsz Grader 作业提交平台，具体截止日期参考平台发布。

- 登录网址：： <http://grader.tery.top:8000/#/login>
- 推荐浏览器： Chrome
- 初始用户名、密码均为学号，登录后请修改



## 附录

Hashpump是hash长度扩展攻击的工具，下载方式如下，本次实验我们没有使用，在长度扩展攻击中，该工具是常用工具。

```
git clone https://github.com/bwall/HashPump
sudo apt-get install g++ libssl-dev
cd HashPump
make
sudo make install
```

关于Hash 长度扩展攻击的参考链接：

<https://blog.skullsecurity.org/2012/everything-you-need-to-know-about-hash-length-extension-attacks>

**请同学们开始实验！**

