海南大学信息学院信息安全系专业课程
《安全协议》
NSPK 协议验证实验报告

姓　　名：
学　　号：
学　　院：
成　　绩：
任课教师：

计算机类课程实验报告

姓　名		主讲教师		专　业	信息安全	班级	
学　号				实验日期	2013-10-9		
课程名称	安全协议			小组成员			

一、实验名称

NSPK(Needham-Schroeder Public-Key Protocol)的验证

二、实验目的

1．加深对 CCITT X.509 (3)协议的理解。

2．掌握软件分析安全协议的基本方法，认识安全协议的基本描述规则，构建完备的知识体系。

三、实验内容及要求

分析 CCITT X.509 (3)协议，并使用 Scyther 协议验证软件对其进行安全验证，完成详尽的报告。

1．建立对 CCITT X.509 (3)协议的认识，在此基础上分析其破解途径。

2．学会使用软件验证安全协议的方法，即协议的代码描述、攻击方法的分析。

3．深入总结实验，进一步发觉安全协议更深层次的知识。

四、实验材料、工具、或软件

Windows XP；Python2.6；Scyther1.1；wxPython2.8-win32-unicode-2.8.12.0-py26

五、实验步骤

本次实验共分四步执行：

第一步：查阅相关资料，系统地认识 Needham-Schroeder 公钥协议；

第二步：思考该协议的验证流程，推敲其攻击破解的方法；

第三步：分析协议的代码描述，并做出详细的解释；

第四步：透过 Scyther 生成的攻击图，分析其具体的攻击流程。

下面按照这四个步骤开始实验并详细叙述。

一、　认识 NS 协议

1. $I \rightarrow R : \{N_I, I\} KP_R$

2. $R \rightarrow I : \{N_I, N_R\} KPi$

3. $I \rightarrow R : \{N_R\} KP_R$

在 Needham-Schroeder 协议中，默认协议双方都从可信服务器中获取了对方的公钥；

1：I 作为协议的发起者，向 R 发送随机数 N_i 及具有新鲜性的随机数 I（也称临时值，nonce）作为挑战，并用 R 的公钥加密；

2：响应者 R 接收并对 I 作出挑战，用 I 的公钥加密已经解密的 N_i 及随机数 N_R 发送给 I；

3：最后，I 向 B 发送用 R 公钥加密的 N_R 。

经过这样一次协议的运行，主体 A 和 B 就建立了一个他们之间的共享秘密 N_R，这个共享秘密可以为他们进行秘密通信确认双方身份时使用。

二、NS 公钥协议攻击

协议的目的是建立协议的发起者 I 与响应者 R 之间的相互认证，并且确保他们在相互通讯，而不会在与某个冒充者通讯。但在 I 与 R 建立认证的过程中，很可能出现入侵者 A，它一方面监听 I 发往 R 的消息，并且截获消息 1，然后 A 可以破坏消息 1，对消息进行篡改，扰乱 I 和 R 的通信。

1.1. $I \rightarrow A:\{N_I, I\}KP_A$

2.1. $A(I) \rightarrow R:\{N_I, I\}KP_R$

2.2. $R \rightarrow A(I):\{N_I, N_R\}KP_I$

1.2. $A \rightarrow I:\{N_I, N_R\}KP_I$

1.3. $I \rightarrow A:\{N_R\}KP_A$

2.3. $A(I) \rightarrow R:\{N_R\}KP_R$

主体 I、主体 R 和攻击者 A，其中 I 作为 NS 公钥协议的发起者，A 作为响应者并假冒 I 和 R 进行通信和欺骗，从而实现对 NS 公钥协议的攻击。协议采用公开密钥系统，N_I、N_R 是 I 和 R 发布的具有新鲜性的随机数（也称临时值，nonce）。攻击过程如下：

首先主体 I 向 A 发送包含 N_I 和自己身份的消息 1.1，并用 A 的公钥 PK_A 加密消息 1.1；A 收到消息并马上对消息进行解密，而后假冒 I 向 R 发送 N_I 和 I 身份（让 R 误认为是和 I 进行通信）的消息1'，并用 R 的公钥 PK_R 进行加密；R 接到消息，并向假冒 I 的 A 发送用 I 的公钥加密的消息 N_I、N_R；而后 A 接到消息 2' 并立即向 I 发送；对协议攻击最后一步，I 向 A 发送 I、R 之间用于通信的共享秘密 N_B 。这样 A 就能得到 I 和 R 之间的共享秘密 N_R，从而实现以后对 I 和 R 通信内容的监听。

根据上述的过程，我们可以发现根据上述攻击可以使 R 不能确认最后一条消息是否来自 I。这是由于 I 从未详细地声明她欲与 R 对话，因此 R 不能得到任何保证 A 知道 R 是她的对等实体。

协议改进如下：

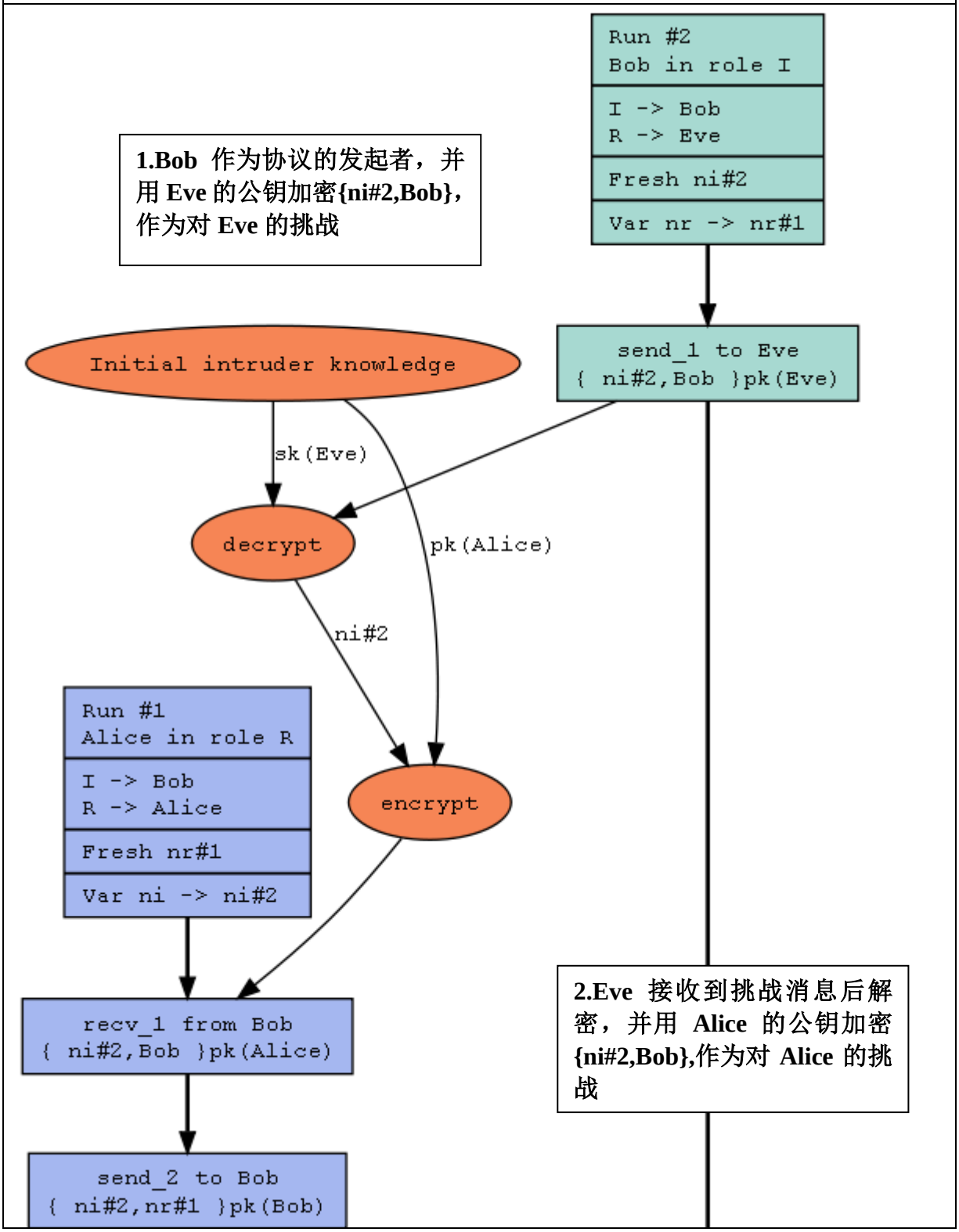
1. $I \rightarrow R:(N_I, I)K\mathrm{P}_R$
2. $R \rightarrow I:(N_I, N_R, R)KP_I$
3. $I \rightarrow R:(N_I)KP_R$

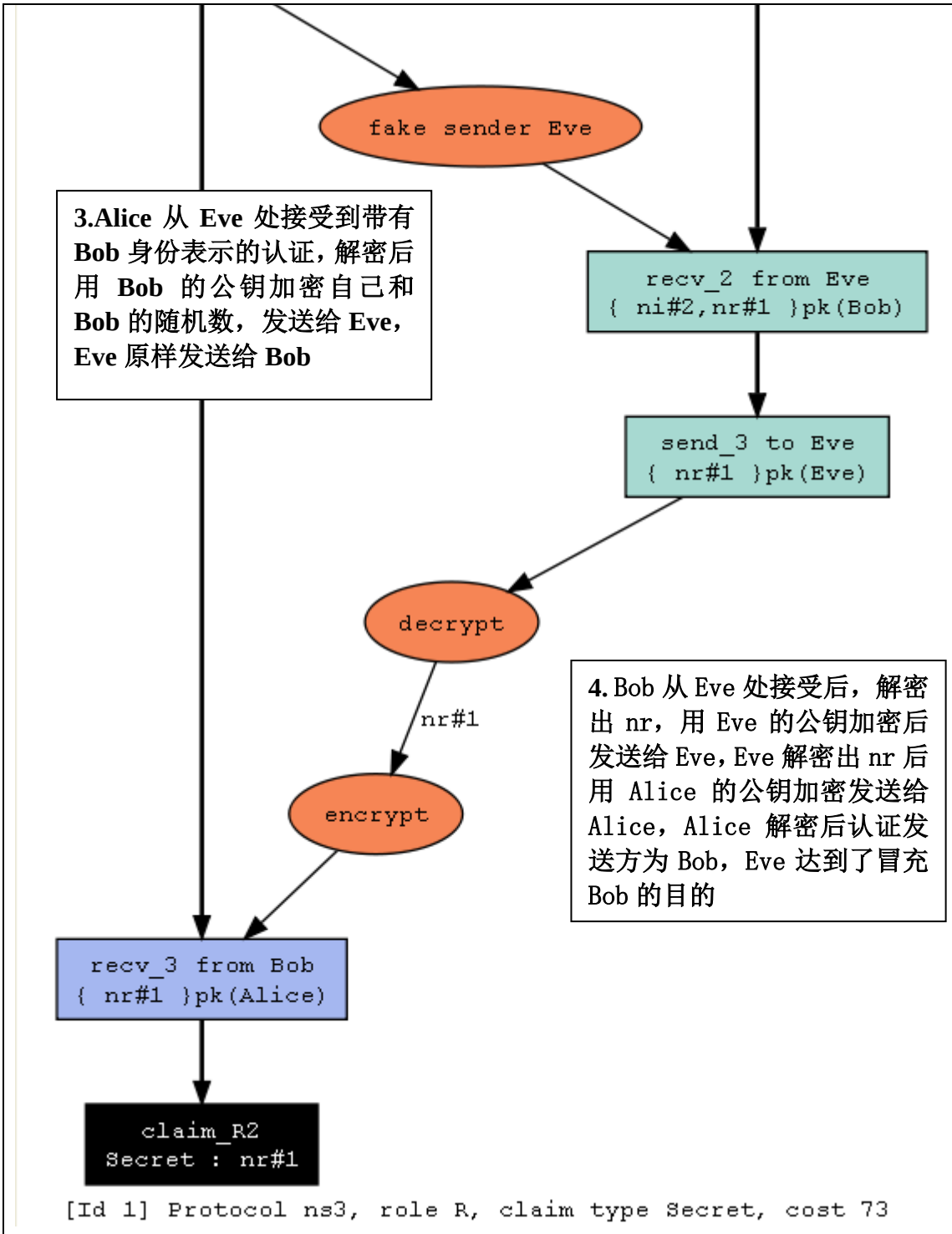
三、分析代码描述

```
/*
 * Needham-Schroeder protocol
 */
// The protocol description
protocol ns3(I,R)      //协议的主体 I、R
{
    role I     //实体 I
    {
        fresh ni: Nonce;        // 产生具有新鲜性的随机数
        var nr: Nonce;
        send_1(I,R, {ni,I}pk(R) );      //发送{ni,I}作为对 R 的挑战
        recv_2(R,I, {ni,nr}pk(I) );     //接受来自 R 的挑战响应{ni,nr}
        claim(I,Running,R,ni,nr);       //声明开始运行协议
        send_3(I,R, {nr}pk(R) );        //I 最后发送共享的密钥{nr}给 R

        claim(I,Secret,ni); //声明 I 发送随机数 ni 的保密性
        claim(I,Secret,nr); //声明 I 已获取秘密随机数 nr
        claim(I,Alive);//声明随机数 I 的新鲜性
        claim(I,Weakagree);  //声明协议的弱一致性
        claim(I,Commit,R,ni,nr);//声明双方已经提交了协议的认证
        claim(I,Niagree);
        claim(I,Nisynch);
    }

    role R          //实体 R
    {
        var ni: Nonce;          // 产生具有新鲜性的随机数
        fresh nr: Nonce;

        recv_1(I,R, {ni,I}pk(R) ); //接受来自协议的发起者 I 的挑战响{ni,I}
        claim(R,Running,I,ni,nr); //声明开始运行协议
        send_2(R,I, {ni,nr}pk(I) ); //发送{ni,nr}作为对 I 的挑战响应
        recv_3(I,R, {nr}pk(R) );//接收到共享的密钥{nr}

        claim(R,Secret,ni); //声明 R 收到随机数 ni
        claim(R,Secret,nr);//声明收到安全的随机数 Nr
        claim(R,Alive); //声明随机数 I 的新鲜性
```

```
        claim(R,Weakagree); //声明协议的弱一致性
        claim(R,Commit,I,ni,nr); //声明双方已经提交了协议的认证
        claim(R,Niagree);
        claim(R,Nisynch);
    }
```

四、分析攻击流程






```
[Id 1] Protocol ns3, role R, claim type Secret, cost 73
```

五、实验存在问题和解决办法

参照学长的实验报告完成了本次实验，对协议重新做了一次全面的分析，对 NS 公钥协议有了更深入的了解。实验软件的更新添加了许多内容，即便协议的大体没有改变，其中较为不解的是“弱一致性”的具体含义，在攻击方面通过协议的分析，及软件的攻击流程图分析有了更深入的了解。

六、协议的改进

```
/*
 * Needham-Schroeder-Lowe protocol
```

```
 */

// The protocol description

protocol nsl3(I,R)
{
    role I
    {
        fresh ni: Nonce;
        var nr: Nonce;

        send_1(I,R, {ni,I}pk(R) );
        recv_2(R,I, {ni,nr,R}pk(I) ); //接受到来至 R  挑战响应，响应中包含了 ni,nr,
和 R 的实体标识 R
        claim(I,Running,R,ni,nr);
        send_3(I,R, {nr}pk(R) );

        claim(I,Secret,ni);
        claim(I,Secret,nr);
        claim(I,Alive);
        claim(I,Weakagree);
        claim(I,Commit,R,ni,nr);
        claim(I,Niagree);
        claim(I,Nisynch);
    }

    role R
    {
        var ni: Nonce;
        fresh nr: Nonce;

        recv_1(I,R, {ni,I}pk(R) );
        claim(R,Running,I,ni,nr);
        send_2(R,I, {ni,nr,R}pk(I) ); //发送{ni,nr,R}作为对 I 的挑战响应(R 为实体 R
的身份标识)
        recv_3(I,R, {nr}pk(R) );

        claim(R,Secret,ni);
        claim(R,Secret,nr);
        claim(R,Alive);
        claim(R,Weakagree);
        claim(R,Commit,I,ni,nr);
        claim(R,Niagree);
        claim(R,Nisynch);
    }
}
```

改进后的攻击过程：

1.1. $I \rightarrow A:\{N_I, I\}KP_A$

2.1.$A(I) \rightarrow R:\{N_I, I\}KP_R$

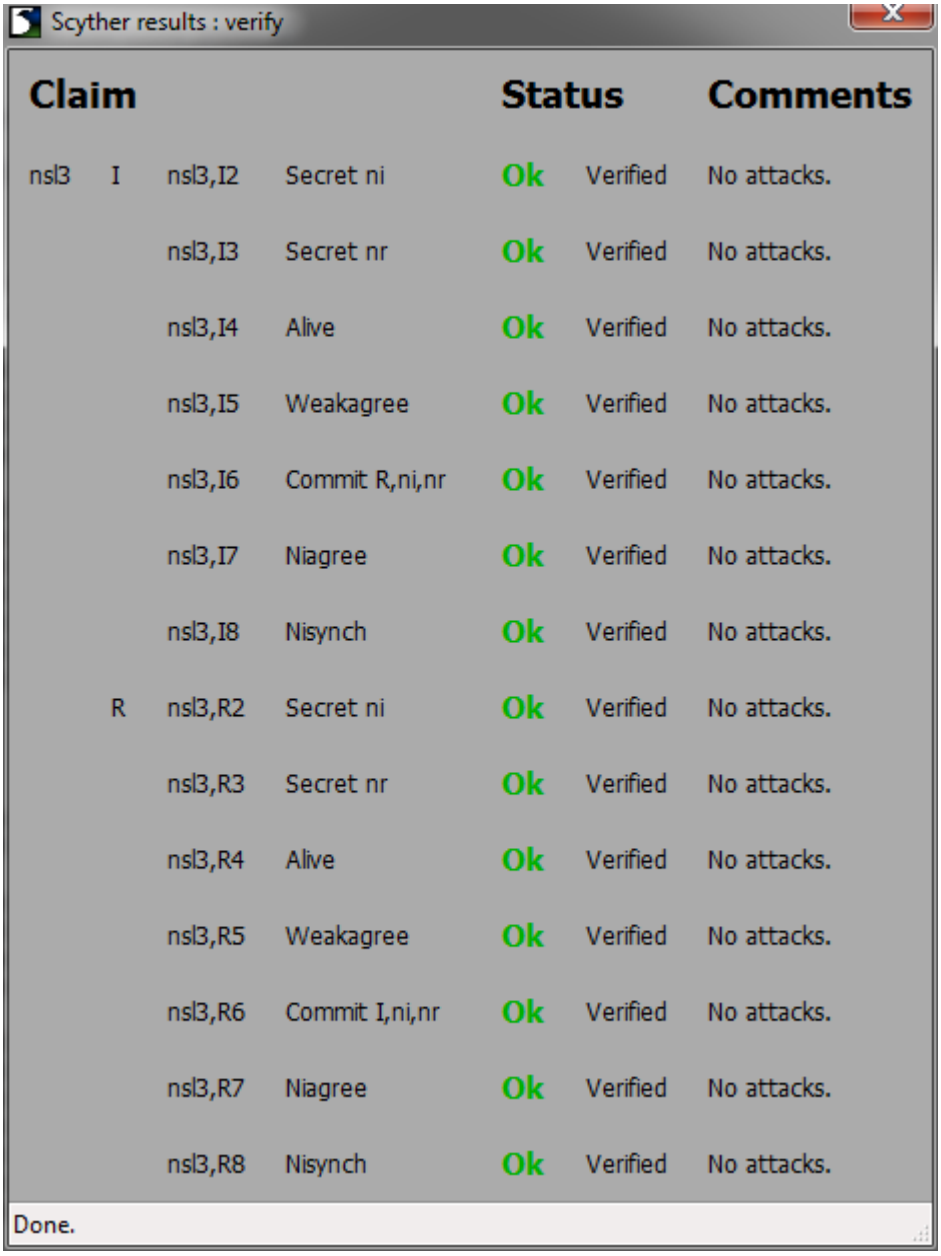
2.2.$R \rightarrow A(I):\{N_I, N_R, R\}KP_I$

1.2.$A \rightarrow I:\{N_I, N_R, R\}KP_I$

1.3.$I \rightarrow A:\{N_R\}KP_A$ //**I 解密 A 发送来的消息，发现身份标识为 R，到此，可认定为 A 想伪造 R 与自己通信，对此不予与回复。**

1.3' 1.3.$I \rightarrow R:\{N_R\}KP_R$ //新的流程

2.3.$A(I) \rightarrow R:\{N_R\}KP_R$ //失效



七、教师评语（或成绩）

教师签字：

年　　月　　日