第 36 卷 第 8 期 计 算 机 工 程 2010 年 4 月
Vol.36 No.8 Computer Engineering April 2010
·安全技术· 文章编号：1000—3428(2010)08—0144—03 文献标识码：A 中图分类号：TP309

基于 Snort 的 IPv6 入侵检测技术

王相林，李蓓蕾
(杭州电子科技大学计算机学院，杭州 310018)

摘 要：针对开源入侵检测系统 Snort 没有提供对 IPv6 协议的 AH 和 ESP 扩展首部支持的问题，提出利用 Snort 检测 ESP 加密报文的解决方案。构造 ESP 检测规则，在 Snort 协议分析模块加入 DecodeESP()函数并添加密钥管理模块，实现 Snort 对 IPv6 报文中 ESP 扩展报头的解析，管理其产生的密钥。给出一种面向 ESP 的入侵检测系统模型，以验证 IPv6 加密通信入侵检测的可行性，并给出实验验证过程。
关键词：入侵检测系统；IPv6 协议；封装安全有效负载

Intrusion Detection Technology in IPv6 Based on Snort

WANG Xiang-lin, LI Bei-lei
(Computer School, Hangzhou Dianzi University, Hangzhou 310018)

【Abstract】Because the free NIDS Snort does not support the analysis of AH and ESP extension header in IPv6 protocol, this paper gives a solution to detect the ESP encrypted IP packets. By constructing ESP testing rules, adding DecodeESP() in Snort protocol analysis module, adding key management module, it solves the problem of anglicizing ESP extension header and the management of key. It builds a model of Intrusion Detection System(IDS) oriented ESP to solve the intrusion detection of encrypted communication in IPv6, and gives the process of the experiment.
【Key words】Intrusion Detection System(IDS); IPv6; Encapsulating Security Payload(ESP)

1 概述

随着 IPv6 协议研究的深入，在 IPv4 向 IPv6 过渡阶段的网络安全领域中出现了许多新课题。IPv6 利用 IPSec 协议解决数据加密及身份认证问题，保证数据在不安全的网络上进行安全传输[1]。IPSec 对于解决针对网络层攻击(如网络侦听、中间人攻击、数据包重放等)是有效的，主要通过 AH 协议和封装安全有效负载(Encapsulating Security Payload, ESP)协议扩展报头来实现。

入侵检测作为一种主动安全防护技术，提供了对内部攻击、外部攻击和误操作的实时保护，在网络系统受到危害之前拦截和响应入侵。目前，基于 IPv6 的入侵检测系统(Intrusion Detection System, IDS)的研究正在进行中。Snort 是一个强大的轻量级网络入侵检测系统，具有很好的扩展性和可移植性，可满足多种应用环境需求。Snort2.8.1 版本开始支持 IPv6 协议，但不支持 IPv6 安全协议 AH 和 ESP 扩展报头。本文改进 Snort 的协议分析模块，实现了 Snort 对 IPv6 报文中的 IPSec 扩展协议 ESP 扩展首部的处理。

2 应用于 IPv6 网络的 Snort 系统

Snort 是基于网络的误用型入侵检测系统，以 Libcap 库函数为基础，用 C 语言编写，主要采用基于规则的网络信息搜索机制，通过对数据包进行协议分析和规则匹配来检测多种不同的入侵行为和探测活动。

2.1 Snort 协议解析机制

Snort 协议解析模块主要由 decode.c 和 decode.h 这 2 个文件组成。它对数据包进行解码分析，并将分析后的数据放到数据结构 Packet 中。Packet 数据结构比较全面地覆盖了数据报文的内容。Snort 的协议解码分析都是围绕 Packet 数据结构进行的。当 Snort 通过网卡捕获数据时，回调函数 PcapProcessPacket()调用 ProcessPacket()函数链接到协议解析程序，由函数指针 grinder 指向具体解析[2]。以太网环境下的 TCP/IP 协议解析流程如图 1 所示。

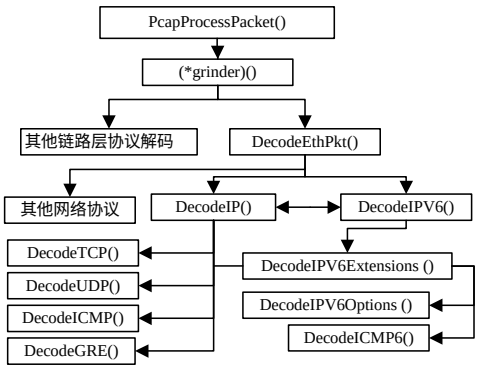

图 1 以太网环境下的 TCP/IP 协议解析流程

可以看出，Snort 解析包括数据链路层、网络层和传输层的解码，其中，(*grinder)()是指针，grinder 指向具体解析；DecodeEthPkt()是指解析采用 Ethernet 的报文。由于 IPv6 是网络层协议，因此需增加对 IPv6 数据包的解析。

2.2 IPv6 协议解析

与 IPv4 协议相比，IPv6 协议简化了报头结构，报头字段由 IPv4 下的 12 个(不包括选项)减少为 IPv6 下的 8 个。IPv6 的报头结构采用的是基本报头加扩展报头的方式，基本报头固定为 8 个字段、40 个字节，用扩展报头的方式实现 IPv4 报头中可选字段的功用，以及 IPv6 协议的其他新特征，例如，用于安全的扩展报头 AH 和 ESP。这样设计加快了路由器等中间节点处理报文的速度，着眼于日后协议扩充的需

作者简介：王相林(1953－)，男，教授，主研方向：网络安全，IPv6 技术，网络性能评价；李蓓蕾，硕士研究生
收稿日期：2009-09-20 **E-mail：**lbl031@yahoo.com.cn

要，便于为 IPv6 协议增加新的功能。

IPv6 报头的这些变化使 IPv6 下的协议解码既具有与 IPv4 相似之处，又体现了自身的特点。Snort 调用 DecodeIPV6() 函数对 IPv6 数据报文进行协议解析，并用 Packet 数据结构存放协议解码分析的结果。它与解析 IPv4 报文类似的是两者都要检查数据报头部的各个字段的合法性。但需要注意的是：由于 IPv6 地址长度扩展为 128 位，使地址的定义不能再用长整型，而改为数组结构来定义；又由于 IPv6 采用扩展头的形式，因此必须检查扩展报头的合法性(如扩展报头的顺序及出现的次数、分段是否正确等)，并调用 DecodeIPV6Extensions() 进行后续的解码[3]。

2.3 IPv6 扩展报头解析

IPv4 的解析是直接通过 DecodeIP()进行的，对 IPv6 的解析不同，它的扩展报头是一个独立的模块，Snort 通过调用 DecodeIPV6Extensions()和 DecodeIPV6Options()函数来实现 IPv6 扩展报头的协议解码。根据 IPv6 协议报头中下一个首部字段的值确定协议包有效载荷，如果是 TCP, UDP, ICMPv6，则调用相应函数进行后续的解码；如果是 Hop-by-Hop, Routing, Fragment 和 Destination Options 扩展报头，则调用 DecodeIPV6Options()函数进行解码，DecodeIPV6Options() 检查扩展报头并产生相应的警报事件，递归调用 Decode IPV6Extensions()进行后续的扩展报头解码。目前所有版本的 Snort 都还没有支持 AH, ESP 扩展报头的解析。

3 面向 ESP 的入侵检测系统模型

为了支持 ESP 扩展的协议解析方法，设计一个面向 ESP 的入侵检测系统结构模型，如图 2 所示，其中，虚线为网络数据；实线为密钥及相关信息。

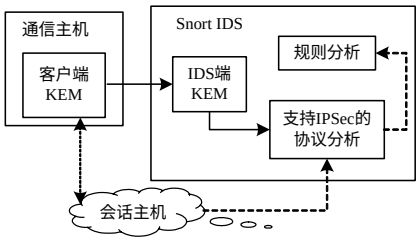

图 2 面向 ESP 的入侵检测系统结构

该模型工作原理如下：

(1)内部网络内的“主机 A”通过 IPSec 和“网内主机 B”或“外网主机 C”等交换动态生成的会话密钥 Key 和安全关联 SA。

(2)主机 A 把 Key 和 SA 使用 IDS 的公钥进行加密然后通过客户端密钥交换模块(Key Exchange Module, KEM)与 Snort IDS 上的 IDS 端 KEM 建立一个连接，把加密的 Key 和 SA、时间戳 t 及连接双方的 IP 都提交给 IDS 端 KEM[4]。

(3)Snort 对网络数据进行分析，在协议分析时数据被加密了，使用加密出来的 Key 和 SA 解密数据包，解密出来的明文进入常规的入侵检测过程。

客户端 KEM 的主要功能有：

(1)公钥管理，保存对应于 IDS 私钥的 IDS 公钥证书，并向加密 Key 和 SA 提交 IDS 公钥。

(2)密钥发送，将产生的 Key 和 SA 经加密后发送给 IDS 端 KEM。

IDS 端 KEM 主要功能有：

(1)接收来自客户端 KEM 提交的 Key 和 SA。

(2)管理私钥，根据私钥生成公钥证书并发送给客户端 KEM。

该模型验证支持 ESP 扩展的协议解析方法的可用性外，通过非对称密钥机制对 ESP 扩展报头所带 Key 进行加密和解密的安全管理，保证了它的安全性。

4 支持 ESP 扩展报头的协议解析

目前的 Snort 还不支持 ESP 扩展报头，只做了简单的默认处理，因此，要实现经 IPSec 协议加密的 IPv6 报文检测就需要修改 Snort 源代码以支持 ESP 扩展报头。

4.1 密钥管理模块

因为 ESP 的加密功能，IPSec 包含密钥交换管理，ESP 涉及到 Key 和 SA，所以设计一个密钥管理模块来负责密钥和安全关联的管理。密钥管理模块的功能是负责 Key 的处理，向 IDS 端 KEM 发出 Key 请求，以及接收 IDS 端 KEM 得到的 Key。该模块根据功能划分为 3 个子模块：Key 处理子模块，接收响应子模块，处理响应子模块。下面给出密钥管理模块的数据存储方法以及该模块工作流程。

用 2 个数据链表分别保存 Key 及相关数据和等待处理的请求，这 2 个链表的定义如下：

(1)链表 1 的定义

```
Data_Link * Key_Buff_Link;   //Key 及相关数据缓存链表
```

相应的 payload 的内部数据结构定义如下：

```
Struct Key_Payload{
Key_Associations key_associations_data; //Key 及相关数据
Time_t time_counter;   //计时器
};
```

(2)链表 2 的定义

```
Data_Link * Hanged_Request_Link;   //等待处理的请求链表
```

相应的 payload 的内部数据结构定义如下：

```
Struct Key_Payload {
Result_Identify_Data request_identifier;   //请求标识
u_int8_t *pkt;   //等待解密的数据包的指针
};
```

Key 管理模块的工作流程如图 3 所示。

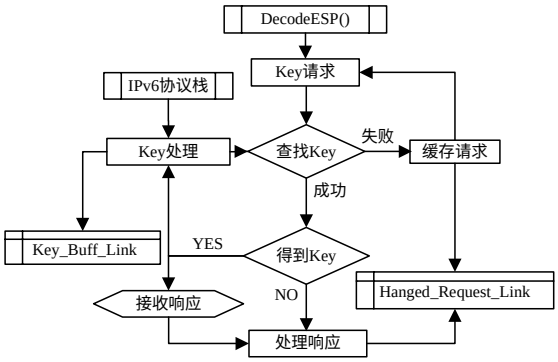

图 3 Key 管理模块的工作流程

密钥管理模块接到来自 DecodeESP()的 Key 请求，到 Key_Buff_Link 里查找，如果查找成功则返回 Key，否则该请求被挂起到 Hanged_Buff_Link；接收响应子模块把接收到的数据提交给处理响应子模块，处理响应子模块把 Key 和 SA 及其他相关信息提交给 Key 处理子模块处理，并把接收的 Key 和 SA 保存到 Hanged_Buff_Link 中以便下次使用；Key 处理子模块还要接收 IPv6 协议栈建立安全连接时产生的 Key 和 SA 及相关信息；Key 处理子模块在保存每个 Key 时还要

为它设置一个密钥生存时间，如果 Key 被使用则重置，如果密钥生存时间到期则删除该 Key 及相关数据。

4.2 ESP 扩展报头解析方案

要支持 ESP 扩展报头的解析，需对 Snort 系统协议解析模块进行改进，让其能够解析 IPSec 协议。解决方法是设计一个 DecodeESP()函数来解析 IPv6 的 ESP 扩展报头。IPv6 扩展报头的解析是用 DecodeIPV6Extensions()函数实现的，因此，在 DecodeIPV6Extensions()函数里调用 DecodeESP()函数来实现 ESP 扩展报头，该函数包含在 decode.c 中。对 Decode IPV6Extensions()函数中添加如下代码：

```
void DecodeIPV6Extensions(u_int8_t next, const u_int8_t *pkt, u_int32_t len, Packet *p)
{…
    switch(next) {        //根据 next 的判断所要解析的扩展首部
        …                 //其他扩展报头
        case IPPROTO_ESP: //ESP 扩展报头
            pc.esp6++;
            DecodeESP(pkt , len, p); //调用 DecodeESP()函数 //解析 ESP 扩展首部
            Return;    …}
…}
```

当 DecodeIPV6Extensions()函数检测到 ESP 扩展报头，则调用 DecodeESP()，函数流程如图 4 所示。

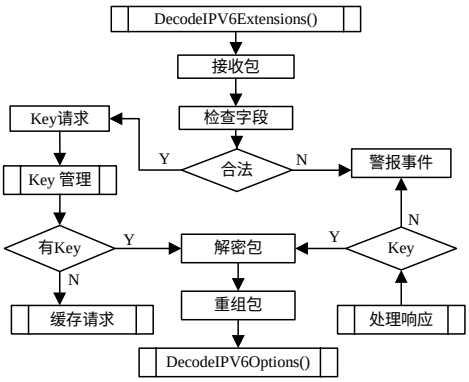

图 4　DecodeESP()函数流程

DecodeESP()函数对报头各字段进行合法性检查，若合法则发出 Key 请求，否则产生警报事件；如果密钥管理模块查找成功则进行协议包解密，否则缓存请求由 IDS 端 Internet KEM 向客户端 KEM 请求 Key(这里的 IDS 端 KEM 和客户端 KEM 同图 2 中的 IDS 端 KEM 和客户端 KEM)；若处理响应子模块提交的结果是 Key 则进行协议包解密，否则产生警报事件；协议包解密完成后进行协议包重组，最后调用 DecodeIPV6Options()函数进行后续的处理。

5　实验与分析

实验采用 2 台直连的主机 A 和主机 B，安装有系统 Red Hat Linux9.0，在 2 台主机上配置好 IPv6 和 IPSec 2 个协议，其中，在主机 A 上安装 Snort2.8.1 以及关联配置，实验的数据都在主机 A 上获得。本文选择 ping 命令来进行实验的连接测试。

实验步骤分析如下：

(1)在主机 A 上设置 Snort 检测规则：alert ip fe80::250:56ff:fec0:2 any->fe80::250:56ff:fec0:1 any\ (msg: "LOCAL IPv6 Link Local test"; sid:2009001)，其中，fe80::250:56ff:fec0:1 是主机 A 的地址；fe80::250:56ff: fec0:2 是主机 B 的地址。该规则表示由主机 B 向主机 A 发送的连接测试进行报警。

(2)在主机 A 上启动 Snort，使其处于网络入侵检测模式。

(3)在主机 B 上对部署 NIDS 的主机 A 进行连接测试：ping6 fe80::250:56ff:fec0:1。

(4)主机 A 上的入侵检测系统的输出模块根据检测结果发生报警，报警日志截图如图 5 所示。结果说明未经优化的 Snort 对由主机 B 向主机 A 发送的未进行 ESP 加密处理连接的数据发生警报。

```
05/07-15:24:01.19581 [**] [1:2009001:0] LOCAL IPv6 Link Local test [**]
[Priority: 0] {IPV6-ICMP} fe80:0000:0000:0000:0250:56ff:fec0:0002 ->
fe80:0000:0000:0000:0250:56ff:fec0:0001
```

图 5　报警日志截图 1

(5)在 2 台主机上建立 Internet 协议安全(IPSec)的安全关联及其他配置，将主机 A 与主机 B 进行通信的数据包进行 ESP 加密传输模式配置。再由主机 B 向主机进行 ping 连接测试，主机 A 上的入侵检测系统并未作出相应的报警。结果说明未经优化的 Snort 无法识别经 ESP 加密的数据包。

(6)在步骤(5)的基础上，将已经配置好优化的 Snort 的主机 C 替换主机 A，其他配置同主机 A，再次进行步骤(5)的操作，结果出现报警，报警日志截图如图 6 所示。结果说明优化的 Snort 对经 ESP 加密的数据包识别并发生警报。

```
05/10-10:44:42.56870 [**] [1:2009001:0] LOCAL IPv6 Link Local test [**]
[Priority: 0] {IPV6-ICMP} fe80:0000:0000:0000:0250:56ff:fec0:0002 ->
fe80:0000:0000:0000:0250:56ff:fec0:0001
```

图 6　报警日志截图 2

以上 3 个结果与设计预期一致，由此可以得出结论：实现 IPv6 加密通信的入侵检测在实践上是可行的。

6　结束语

本文在分析 IPv6 报头解析的基础上，根据 IPv6 协议扩展报头 ESP 的特征，制定检测 ESP 加密报文的解决方案，提出面向 ESP 的入侵检测系统模型，并进行了验证。在解决加密问题后，下一步还要对入侵检测技术的检测速度、漏检率和误检率等性能做进一步探讨，程序关键部分的代码优化也需进一步改进。

参考文献

[1] 王相林. IPv6 核心技术[M]. 北京: 科学出版社, 2009.

[2] 谢丽霞, 杨宏宇. Snort 报文检测及解析机制分析[J]. 航空计算技术, 2005, 12(4): 109-110.

[3] 李振强, 徐一元, 马　严. 基于 Snort 的 IPv6 入侵检测系统的研究与实现[J]. 电信科学, 2005, 21(8): 32-35.

[4] Kent S, Atkinson R. IP Encapsulating Security Payload(ESP)[S]. RFC 4206, 1998.

编辑　顾姣健