面向 OSPF 脆弱点的分节点污染方法研究

周季璇,顾巧云,凤　丹
(江南计算技术研究所,江苏 无锡 214083)

摘　要:为了研究 OSPF 协议某脆弱点的污染范围和效果,对 OSPF 协议的安全机制和已知脆弱点进行了研究,分析了对当前 OSPF 协议危害极大的一种脆弱点。针对该脆弱点的污染范围和效果,提出了一种节点分类和脆弱点分节点污染方法,分析总结出在已知源节点和目标节点的前提下污染路径的生成树确定方法,并利用分节点污染方法在 GNS3 平台上对多区域自治域进行了仿真测试和分析,对比总结了网络拓扑和目标路由器位置的选择变化对整个自治域网络污染范围和效果的影响。仿真实验表明,面向 OSPF 协议某脆弱点的分节点污染方法能够有效分析出整个网络拓扑以及目标节点位置对最终污染范围和效果的影响,有助于对安全网络拓扑设计。最后针对该脆弱点及其污染特征,提出了相应的防范措施。

关键词:开放最短路径优先协议;脆弱点;链路状态宣告;污染路径
中图分类号:TP393　　**文献标识码**:A　　**文章编号**:1673-629X(2018)05-0122-05
doi:10.3969/j.issn.1673-629X.2018.05.028

Research on Pollution Method with Diverging Nodes Injected Based on OSPF Vulnerability

ZHOU Ji-xuan, GU Qiao-yun, FENG Dan
(Jiangnan Institute of Computing Technology, Wuxi 214083, China)

Abstract: In order to research the pollution range and effect of a weak point of OSPF, the security mechanism of OSPF protocol and the known vulnerability are studied, and the current OSPF protocol is widely concerned with a vulnerable point. For this, we propose a kind of node classification and pollution method with diverging nodes injected. Then a method for determining the spanning tree of pollution path on the premise of known source node and target node is summarized with analysis. The pollution method with diverging node injected is used on GNS3 platform for simulation test and research in multi-area autonomous system. The comparison summarizes the effect of the network topology and selection of target router location on the network pollution range. Simulation experiments show that the pollution method with diverging node injected for OSPF protocol can effectively analyze the entire network topology and the target node location for the effects of the pollution range, helpful for the design of security network topology. Finally, we put forward the countermeasures for the vulnerability and its pollution characteristics.

Key words: OSPF; vulnerability; LSA; pollution path

0　引　言

近年来,路由协议的缺陷[1]导致各种针对路由协议的脆弱性利用呈现增长的趋势,因此路由协议的安全研究成为一项十分急迫的任务[2]。OSPF 协议是目前应用广泛的内部网关路由协议,其安全性研究一直受到业界学者的广泛关注,近年来也有不少研究成果。文献[3-5]对 OSPF 协议脆弱性被利用后的检测恢复以及防范进行了研究,文献[6-8]进一步分析研究了 OSPF 协议的安全机制。OSPF 协议在协议细节、认证和配置等方面的相关脆弱点也陆续被发现[9]。Gabi Nakibly 团队在 2013 年公布了一个关于 OSPF 协议的新型脆弱点并且提出了该脆弱点的利用方法[10],同时在自治域只有一个骨干区域的情况下进行了实验验证,得出只需要发送一个包到任意位置的目标路由器就能持久改变该路由器的链路状态数据库,控制其路由表并且污染整个自治域内的其他路由器的结论。国内外也有文献[11-13]针对该脆弱点进行了分析研究,但没有对网络拓扑特征以及目标路由器在网络拓扑中的位置对污染效果的影响进行深入探讨。

对此,文中在 Gabi Nakibly 团队的研究基础上做

收稿日期:2017-06-13　　**修回日期**:2017-10-18　　**网络出版时间**:2018-02-08
基金项目:国家自然科学基金(91430214)
作者简介:周季璇(1993-),女,硕士研究生,研究方向为软件工程;顾巧云,高级工程师,研究方向为软件工程。

了进一步探索,在自治域分为多个区域的情况下,提出一种 OSPF 脆弱点分节点污染方法,并且对该方法进行了仿真测试。总结出网络拓扑和目标路由器位置选择对污染范围和效果的影响,并提出了利用该脆弱点的污染路径生成树确定方法,最后针对该脆弱点,特别是在分节点污染情况下的特征提出了防范措施。

1　OSPF 安全机制及脆弱点

OSPF 协议有三个内建的安全机制,分别是可靠泛洪与自反击机制,层次路由和信息隐藏,程序性检测及约束,它们使得 OSPF 协议更加强壮,攻击更加困难[14]。其中,可靠泛洪与自反击机制是最重要的一条,尤其是自反击机制,是指一旦某节点路由器收到了一条 Advertising Router 等于自身 ID 的 LSA 实例,便会认为该 LSA 是自己的实例,而这个实例比该路由器中对应的实例更新,那么该路由器就会立即泛洪一条比接收到的 LSA 更新的实例[15]。因为自反击机制的存在,攻击者冒充网络中路由器发送的恶意 LSA 很快就会被更新的正确 LSA 所覆盖,因此,为了能够有效修改链路状态数据库中的 LSA 项,攻击者必须持续不断地发送攻击包,成本较高。

GabiNakibly 团队经过研究发现,OSPF 协议进程在将 LSA 放入链路状态数据库时,仅以 Link State ID 项来标识唯一的 LSA 实例并且不对 Link State ID 和 Advertising Router 项是否相等进行校验,而对于 Router-LSA(描述产生路由器接入某区域的接口状态),这两项必须相等[13,16]。因此,假设发送一条包含恶意 Router-LSA 的数据更新包给某一个目标路由器,满足以下条件:

(1) Link State ID = 目标路由器的 ID;

(2) Advertising Router ≠ Link State ID,且不等于网络内任何一个路由器的 ID;

(3) LSA 的序列号(即 sequence number 项)大于该路由器对应的有效 LSA 的序列号。

那么,此恶意 Router-LSA 就能成功避开 OSPF 的自反击机制替换原来正确有效的 LSA,进入目标路由器的链路状态数据库,并且通过泛洪机制扩散出去,污染其他的路由器,参与到路由表的计算中。由于目标路由器自身产生的 Router-LSA 信息从链路状态数据库中清除了,目标路由器的路由功能基本失效。此外,如果在恶意 Router-LSA 中精心构造虚假链接就能够欺骗被污染路由器,造成流量黑洞或路径劫持等诸多危害,被污染的路由器越多,危害范围越大。

发现这一脆弱点的 Gabi Nakibly 等在文献[10]中表示,利用该脆弱点能影响到整个自治域内所有路由器而对目标路由器在自治域内的位置不作要求。文中在其基础上进行了进一步的探索,提出了 OSPF 脆弱点分节点污染方法,在自治域分为多个区域的情况下,对该方法进行了仿真测试,总结出网络拓扑和目标路由器节点位置对整个自治域污染范围和效果影响,并且提出一种用于确定污染路径及范围的生成树算法。

2　脆弱点分节点污染方法

2.1　节点分类方法

只考虑一个自治域的情况,在文献[15]中,根据路由器的位置,可以将路由器分成内部路由器(IR)和区域边界路由器(ABR)。在此基础上,文中对每个位置的路由器节点进一步细分。如图 1 所示,R_1,R_2,R_3,R_6是区域 0 的 IR 节点,R_7是区域 1 的 IR 节点,R_8是区域 2 的 IR 节点,R_4和 R_5是区域 0 和区域 1 的 ABR 节点,R_4也是区域 0 和区域 2 的 ABR 节点。

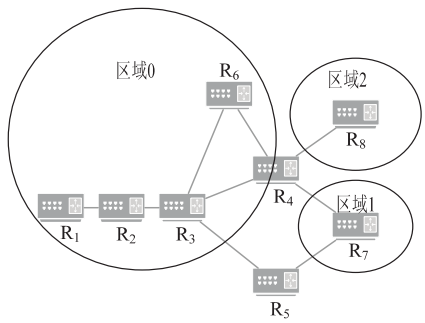

图 1　网络拓扑

现根据每个节点的邻居节点的种类,继续将节点进行细分。以 n-xIR-yABR 的子类别进行分类,其中 n 表示该节点有 n 个邻居节点($n=1$ 时该节点为叶子节点),xIR 表示 x 个 IR 节点的邻居,yABR 表示 y 个 ABR 节点的邻居,$x+y=n$。以图 1 为例,R_1的类别是 IR,子类别是 1-1IR,R_4的类别是 ABR,子类别是 4-4IR,其他节点以此类推。

2.2　分节点污染方法

由于恶意 Router-LSA 包只能通过源节点向邻居发出,所以在测试时选取邻居数量及类型最多的节点为源节点,以它的某一个邻居为目标节点,向目标节点发送恶意 Router-LSA 包,再利用同样的方法选择不同的邻居为目标节点进行多次污染测试,对比分析结果。通过该方法能在控制一个源节点的情况下测试到更多数量和类型的目标节点,从而得到目标节点位置选择对污染结果的影响规律。

分节点污染方法的步骤如下:

(1)根据提出的节点分类方法对网络拓扑中每个路由器节点确定子类别(n-xIR-yABR);

(2)选择 n,x,y 均比较大的节点作为源节点;

(3)每次选择源节点的某一个邻居,利用脆弱点进行测试,记录污染结果,直至所有邻居都被作为目标

节点进行过测试;

(4)对比分析不同节点作为目标节点时的污染结果,总结规律。

以图 1 为例,采取分节点污染方法对其进行测试分析。选取 R_3(子类型 4-2IR-2ABR)为源节点,通过 R_3 分别以 R_2,R_4,R_5,R_6 为目标节点进行多次发包测试,针对这些不同类型和位置的目标节点分析总结每次测试的污染范围和效果。

2.3 污染路径生成树确定方法

目标节点接收到恶意 Router-LSA 后,将其替换掉原本正确有效的 LSA 进入链路状态数据库的 Router Link States 项,并且通过泛洪机制将其从一些特定接口扩散到其他节点。目标节点会把恶意 Router-LSA 从接收到该条 LSA 的接口所在区域的其他所有接口泛洪出去,也就是说,恶意 LSA 只能在某一个区域里泛洪而不会泛洪进其他区域,值得注意的是,对于目标节点而言,恶意 LSA 将不会从接收接口泛洪。此外,当目标节点是 ABR 时,除了会在接收恶意 Router-LSA 的同区域内不作修改地泛洪恶意 LSA 之外,还会向其他区域泛洪描述区域网络状况的 Summary-LSA(由 ABR 产生,向区域汇总路径)的更新信息。Summary-LSA 的泛洪机制和 Router-LSA 类似,只能在区域内泛洪且不从接收的接口进行泛洪。接收到 Summary-LSA 的其他区域节点的 Summary Net Link States 将会被污染。

根据以上对 OSPF 脆弱点以及泛洪机制的分析,文中提出了一种恶意 LSA(假设宣告空链接)污染路径的确定方法。首先,假设 R_n 是源节点,它的某邻居节点 R_t 是目标节点,发送恶意 Router-LSA 包的接口在区域 a,区域 a 以外的所有区域称为其他区域。按照以下方法生成污染路径树。

(1)以 R_n 为根节点,子节点为 R_t(接收恶意 Router-LSA);

(2)R_t 的子节点根据以下规则确定:R_t 是 IR 时,其子节点是区域 a 内除了 R_n 的所有邻居(接收 Router-LSA);

(3)R_t 是 ABR 时,其子节点是区域 a 内除了 R_n 的所有邻居(接收 Router-LSA)以及其他区域的所有邻居(接收 Summary-LSA)。

完成前两步之后,网络中所有节点接收到有效的污染更新消息 x 并且会将其泛洪到的子节点由函数 $f(x)$ 确定。

$$f(x)=\begin{cases}\text{区域 a 内除了父节点的所有邻居节点} \\ (x=\text{Router}-\text{LSA}) \\ \text{接收 } x \text{ 的区域内除了父节点的所有邻居节点} \\ (x=\text{Summary}-\text{LSA})\end{cases}$$

根据以上方法构造生成一棵污染路径生成树,从根节点(不包括根节点)到叶子节点即是一条污染路径,如果路径中存在某个节点是非区域 a 的 IR,那么该条路径是 Summary-LSA 的泛洪路径(不包括作为根节点的子节点时的目标节点),否则是恶意 Router-LSA 的泛洪路径。

以图 1 中的 R_3 为源节点,R_4 为目标节点为例,利用上述方法构造的污染路径生成树如图 2 所示。

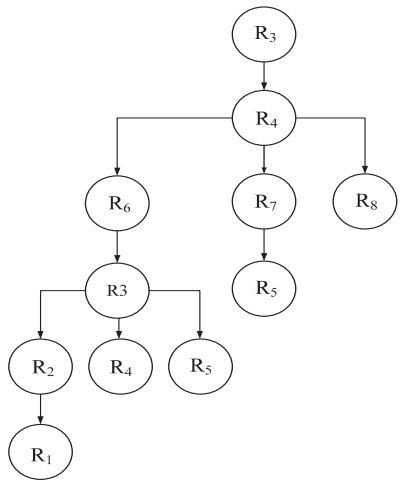

图 2 污染路径生成树

从图 2 中可以看出,恶意 Router-LSA 污染路径是 $R_4\rightarrow R_6\rightarrow R_3\rightarrow R_2\rightarrow R_1$,$R_4\rightarrow R_6\rightarrow R_3\rightarrow R_4$,$R_4\rightarrow R_6\rightarrow R_3\rightarrow R_5$。而 Summary-LSA 污染路径是 $R_7\rightarrow R_5$,R_8。

3 仿真测试

3.1 测试环境

该脆弱点适用于一切遵从 OSPF 协议规范的路由器,主导着路由器市场的 Cisco 也未能幸免。文中选取 Cisco 型号为 c7200(15.0)的路由器,利用脆弱点分节点污染方法进行了仿真测试。由于实物成本较高,在此采用 GNS3 模拟器对实验环境进行仿真研究。

3.2 测试拓扑

采用的实验拓扑如图 1 所示。根据分节点污染方法,选择 R_3 作为测试源节点。在测试时,使 R_3 与本地云相连接,伪装成 R_3 给目标节点(R_3 的邻居节点)发送带有恶意 Router-LSA 的协议包。

3.3 测试方法

利用 Scapy,通过 R_3 发送恶意 Router-LSA 包,每次选择一个目标节点(R_2,R_4,R_5,R_6),进行多次测试。将该条 LSA 的 Link ID 设置成目标路由器的 ID,Advertising Router 设置成任意不属于该网络中任何路由器的 ID,并且序列号大于对应的有效 LSA 的序列号。同时,在恶意 LSA 里没有宣告任何链接。以目标节点是 R_4 时为例,测试代码如下:

```
Fromscapy. all import *
```

```
Load_contrib("ospf")
R4_FALSE_LSA = IP(src = attacker_source_ip, dst = victim_destination_ip) \
/OSPF_Hdr(src = attacker_router_id) \
/OSPF_LSUpd(lsalist = [OSPF_router_LSA(options = 0x22, type = 1, id = victim_router_id, adrouter = false_adv_router, seq = seq_num, linklist = [])])
send(R4_FALSE_LSA, iface = "eth0")
```

3.4　测试结果

测试之后,恶意 Router-LSA 替换掉原本有效的 LSA 进入目标节点的链路状态数据库,并逐步泛洪至其他节点,从而对一定范围的网络造成了污染。图 3 和图 4 是目标节点为 R_4 时,R_4 和 R_7 在测试前后链路状态数据库的变化图。可以看到,对路由器链路状态数据库的污染确实主要表现在对路由器 Router Link States 项以及对 Summary Net Link States 项的篡改上。

OSPF Router with ID(50.0.0.1)　(process ID 1)

Router Link States(Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
18.0.0.1	18.0.0.1	56	0x80000002	0x000FA2	2
18.0.0.2	18.0.0.2	57	0x80000002	0x00B130	1
40.0.0.1	40.0.0.1	54	0x80000002	0x00F8BA	1
50.0.0.1	50.0.0.1	49	0x80000002	0x0096A4	2
55.0.0.1	55.0.0.1	50	0x80000002	0x002CAA	2
192.168.16.8	192.168.16.8	54	0x80000002	0x0003AA	5

OSPF Router with ID(50.0.0.1)　(process ID 1)

Router Link States(Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
18.0.0.1	18.0.0.1	56	0x80000002	0x000FA2	2
18.0.0.2	18.0.0.2	57	0x80000002	0x00B130	1
40.0.0.1	40.0.0.1	54	0x80000002	0x00F8BA	1
50.0.0.1	11.11.11.11	6	0x80000004	0x007A12	4
55.0.0.1	55.0.0.1	50	0x80000002	0x002CAA	2
192.168.16.8	192.168.16.8	54	0x80000002	0x0003AA	5

图 3　R_4 数据库中 Router Link State 项的变化

Summary Net Link States(Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.0.0.0	40.0.0.1	115	0x80000001	0x00AB5C
10.0.0.0	50.0.0.1	164	0x80000001	0x0047B7
16.0.0.0	40.0.0.1	115	0x80000001	0x005DA4
16.0.0.0	50.0.0.1	115	0x80000001	0x0003F4
18.0.0.0	40.0.0.1	105	0x80000001	0x004DB1
18.0.0.0	50.0.0.1	106	0x80000001	0x00F202
20.0.0.0	40.0.0.1	153	0x80000001	0x001FDF
20.0.0.0	50.0.0.1	116	0x80000001	0x00CE25
32.0.0.0	40.0.0.1	106	0x80000001	0x00965A
32.0.0.0	50.0.0.1	165	0x80000001	0x0028C0
50.0.0.0	40.0.0.1	106	0x80000001	0x00AB33
50.0.0.0	50.0.0.1	165	0x80000001	0x003D99
55.0.0.0	40.0.0.1	116	0x80000001	0x00607A
55.0.0.0	50.0.0.1	116	0x80000001	0x0006CA
192.168.0.0	40.0.0.1	116	0x80000001	0x007C2C
192.168.0.0	50.0.0.1	116	0x80000001	0x00227C

Summary Net Link States(Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.0.0.0	40.0.0.1	115	0x80000001	0x00AB5C
16.0.0.0	40.0.0.1	115	0x80000001	0x005DA4
18.0.0.0	40.0.0.1	105	0x80000001	0x004DB1
20.0.0.0	40.0.0.1	153	0x80000001	0x001FDF
32.0.0.0	40.0.0.1	106	0x80000001	0x00965A
32.0.0.0	50.0.0.1	165	0x80000001	0x0028C0
50.0.0.0	40.0.0.1	106	0x80000001	0x00AB33
55.0.0.0	40.0.0.1	116	0x80000001	0x00607A
192.168.0.0	40.0.0.1	116	0x80000001	0x007C2C

图 4　R_7 数据库中 Summary Net Link State 项的变化

分析**因方数据**的结果可知,外部注入的恶意 Router-LSA 成功避开了自反击机制替换掉原来正确有效的 Router-LSA,进入到目标路由器 R_4 的链路状态数据库,篡改了数据库中的 Router Link States(Area 0)项。Link ID 是 50.0.0.1 的表项,ADV Router(即上文的 Advertising Router)已经由原本 R_4 的 ID(50.0.0.1)变成了恶意 LSA 中宣告的虚假信息 11.11.11.11。而 R_7 的链路状态数据库中,Summary Net Link States(Area 1)项也遭到了篡改。由 ABR 之一的目标节点 R_4(50.0.0.1)宣告的所有区域 0 内的网络状态信息均被删除了,只保留了由 R_4 宣告的区域 2 的网络状态以及同为 ABR 的 R_5(40.0.0.1)宣告的网络状态。

选择不同类型和位置的目标节点进行测试,得到的污染效果如表 1 所示。RLS 即为路由器链路状态数据库中的 Router Link States 项,SNS 即为数据库中的 Summary Net States 项。对比表 1 由实验得出的受污染节点的范围和图 2 通过污染路径生成树确定方法理论分析出的结果,可以看出,实验测试结果与理论分析的污染结果完全相同,证实了污染路径生成树方法的有效性。因此当该脆弱点被利用后,可以通过该方法迅速确定整个网络拓扑中所有路由器节点的受污染情况,从而及时做出应对策略。

表 1　不同目标路由器节点的污染范围

目标路由器	类别	RLS 改变	SNS 改变
R_2	IR(2-2IR)	R_1,R_2	无
R_4	ABR(4-4IR)	R_1,R_2,R_3,R_4,R_5,R_6	R_5,R_7,R_8
R_5	ABR(2-2IR)	R_5	R_4,R_7
R_6	IR(2-1IR-1ABR)	R_1,R_2,R_3,R_4,R_5,R_6	无

3.5　测试结论

文中利用提出的 OSPF 脆弱点分节点污染方法进行仿真测试,每次测试时向不同类型和位置的目标节点注入一条宣告空链接的恶意 Router-LSA,证实了目标节点在网络拓扑的节点位置对最终污染范围和效果存在很大的影响,并且得出以下结论:

(1)提出的污染路径生成树方法可以有效确定污染范围和路径。

(2)发包的源节点无法从目标节点接收到恶意 LSA,因此只有在拓扑成环时源节点才能收到恶意 LSA。例如当目标节点是 R_4 时,R_3 接收到恶意 LSA 的过程是 $R_3 \rightarrow R_4 \rightarrow R_6 \rightarrow R_3$,而不是直接从 R_4 接收。

(3)Summary Net Link States 的更新信息只能在一个区域内泛洪。依然以文中的仿真测试为例,当 R_4 是目标节点时,R_4 生成的 Summary-LSA 更新信息泛洪至区域 1 的 R_7,修改了 R_7 的 Summary Net Link States 项,R_7 再将其泛洪到作为 ABR 的 R_5,但没有通过 R_5 泛洪到区域 0 的 R_3,即没有能够跨过 ABR 到达

其他区域。

(4)只有当目标节点是 ABR 时,才会影响到其他节点路由器的 Summary Net Link States 项,ABR 自身的 Summary Net Link States 项不会因为恶意 Router-LSA 而改变。

4 防范措施

针对该脆弱点,一般可以采取协议包加密认证,加大攻击者攻击难度;在实现 OSPF 协议时对 LSA 实例增加对 Router-LSA 的 Link State ID 和 Advertising Router 字段是否相等的校验,从本质上消除脆弱点等措施预防该脆弱点被利用。

文中利用脆弱点分节点污染方法对脆弱点进行了仿真测试,基于对测试结果的分析,提出以下防范措施以减小该脆弱点被利用之后的影响。

(1)设计链路状态数据的检测和预警机制[12],在其突然发生巨大变化时发出警报并及时采取重启目标节点等有效手段进行防护。

(2)加强对关键节点的防护。首先该脆弱点的利用一定需要控制一个路由器节点。对于 n-xIR-yABR 类型节点,n,x,y 三个值越大,说明该节点作为被控制的源节点或者目标节点时,可以污染到的节点数目和种类越多,危害也越大。因此要加强对该类节点防护。以图 1 为例,R_3,R_4等节点都应该是重点防护节点。

(3)改进网络拓扑的设计。尽可能解开拓扑中的环形。仍然以图 1 测试拓扑为例,解开 R_3和 R_6之间链接,把环形拓扑打开,就能极大地减小污染范围。改进拓扑后仍以 R_3为源节点进行仿真测试,结果如表 2 所示。

表 2 拓扑改进后不同目标节点的污染范围

目标	类别	RLS 改变	SNS 改变
R_2	IR(2-2IR)	R_1,R_2	无
R_4	ABR(4-4IR)	R_4,R_6	R_5,R_7,R_8
R_5	ABR(2-2IR)	R_5	R_4,R_7

R_6的子类型发生了改变,也无法再成为目标节点,其他节点虽然子类别没有变化,但是由于网络拓扑的变化,污染范围也明显减小。

5 结束语

OSPF 协议发展至今,已成为一项比较成熟和完善的路由协议,但是仍然不能忽略它可能存在的安全隐患和威胁。针对 OSPF 协议某脆弱点的污染特征,提出分节点污染方法和污染路径生成树的确定方法,并通过仿真测试验证了这两种方法的有效性。在构造恶意 Router-LSA 时宣告的是空链接,将来可考虑在恶意 Router-LSA 宣告不同形式的链接时,对污染效果的影响进行更深层次的探索和研究。

参考文献:

[1] 梅鸿翔. OSPF 路由协议的安全性评测研究[D]. 成都:电子科技大学,2010.

[2] 王先培,文云冬,高志新,等. OSPF 路由协议的脆弱性分析[J]. 武汉大学学报:工学版,2004,37(3):98-101.

[3] SANGROHA D,GUPTA V. Analyzer router:an approach to detect and recover from OSPF attacks[C]//International symposium on security in computing and communication. Berlin:Springer,2014:370-378.

[4] LI Meng,JING Quanliang,YAO Zhongjiang,et al. On the prevention of invalid route injection attack[C]//International conference on intelligent information processing. Hanzhou,China:[s. n.],2014:294-302.

[5] SOSNOVICH A,GRUMBERG O,NAKIBLY G. Finding security vulnerabilities in a network protocol using parameterized systems[C]//International conference on computer aided verification. [s. l.]:[s. n.],2013:724-739.

[6] WANG Minghao. The security analysis and attacks detection of OSPF routing protocol[C]//7th international conference on intelligent computation technology and automation. Changsha,China:IEEE,2014:836-839.

[7] DIWAN D,NARANG V K,SINGH A K. Security mechanism in RIPv2,EIGRP and OSPF for campus network-a review[J]. International Journal of Computer Science Trends and Technology,2017,5(2):399-404.

[8] SHEN N,AGGARWAL R,SHAFFER S. Extensions to OSPF for advertising optional router capabilities[J]. Work in Progress,2007,11(3):82-89.

[9] 蔡昭权. OSPF 路由协议的攻击分析与安全防范[J]. 计算机工程与设计,2007,28(23):5618-5620.

[10] NAKIBLY G,MENAHEM E,WAIZEL A,et al. Owing the routing table part2[R]. USA:Black Hat,2013.

[11] 夏云峰. 基于 OSPF 路由协议的路由欺骗分析[D]. 南京:东南大学,2014.

[12] 周 轩,王永杰,覃志波. OSPF 协议漏洞机理及其防范措施[J]. 指挥信息系统与技术,2015,6(5):40-45.

[13] NAKIBLY G,MENAHEM E. OSPF vulnerability to persistent poisoning attacks:a systematic analysis[C]//Proceedings of the 30th annual computer security applications conference. New Orleans,Louisiana,USA:ACM,2014:336-345.

[14] 钟廷龙,李 鑫,郭云飞. OSPF 路由协议安全性分析[J]. 微计算机信息,2005,24:15-17.

[15] MOY J. OSPF version 2[S]. [s. l.]:IETF,1998.

[16] NAKIBLY G,KIRSHON A,GONIKMAN D,et al. Persistent OSPF attacks[C]//Proceedings of the 19th annual network and distributed system security symposium. [s. l.]:[s. n.],2012.