Novel Attacks in OSPF Networks to Poison Routing Table

Yubo SONG*†, Shang GAO†, Aiqun HU*, Bin XIAO†
*School of Information Science and Engineering, Southeast University
†Department of Computing, The Hong Kong Polytechnic University
songyubo@seu.edu.cn, cssgao@comp.polyu.edu.hk, aqhu@seu.edu.cn, csbxiao@comp.polyu.edu.hk

***Abstract*—Link State Advertisement (LSA) reflects the current status of all incident links of a router in an Autonomous System (AS). A fake LSA with false link status information will pollute the view of the network topology on routers. In this paper, we present two novel attacks that inject malicious Link State Advertisements (LSAs) to modify the routing tables: adjacency spoofing and single path injection. Adjacency spoofing attack makes attacker access to routing networks by disguising as a legitimate router. Single path injection attack evades the "fight-back" mechanism and affects routing advertisements of routers. Unlike existing LSA injection attacks, which need to be launched by malicious routers, a common host can launch these attacks and control the transmission path of data traffic in an AS. Simulation and real-world experiment results show that these two attacks can efficiently modify the routing tables of routers, and further lead to DNS spoofing, phishing Website, eavesdropping, and man-in-the-middle attacks. Furthermore, we also implement a security vulnerability detection system to detect the existing vulnerabilities of routing protocol deployed in real-world routers.**

I. Introduction

Routing protocol specify how routers communicate with each other, disseminating information that enables them to select routes between any two nodes on a computer network. The most widely used class of the routing protocols is link-state routing protocol. Examples of these protocols include Open Shortest Path First Protocol (OSPF) [1], Optimized Link State Routing Protocol (OLSR) [2] and Intermediate System to Intermediate System (IS-IS) [3]. These routing protocols are usually deployed on a single Autonomous System (AS). Each AS is administered by a single authority, such as a large organization or an Internet Service Provider (ISP). It allows all routers in the AS to build the same representation of the AS's network topology.

In link-state routing, every node keeps a "map" of the entire network, which is used to compute the shortest paths to all destinations. Each node contributes to this global view by distributing link-state information periodically. Each router composes a list of all links to neighboring routers and their costs. This list termed as Link State Advertisement (LSA) reflects the current status of all the incident links of a given node. Each LSA is flooded throughout the AS, and every router compiles a database of the LSAs from all other routers in the AS. Thus, each router can obtain a complete view of the AS topology, and is able to calculate the least cost paths to every other router in the AS based on shortest paths finding algorithm such as Dijkstra's algorithm adopted in OSPF networks. As a result, the router's routing table is formed.

The integrity protection of the data packet is considered with modern cryptographic algorithms [4–7]. However, the key is used directly in these protocols as specified. Related key attacks, such as those described in [4] are possible. Furthermore, apparently the key management implementations have trouble in practice. For few inner-network administrators prefer to configure the keys on routers manually one by one.

The most common attack against link-state routing protocol is LSA falsification, in which an attacker advertises a fake LSA with false link information on the behalf of routers in the AS. In link-state routing networks, a router periodically advertises a new instance of each LSA throughout the AS. Every LSA has a sequence number which increases with every new advertised instance. Besides, more recent LSA instances always take precedence over older instances. In LSA falsification attack, the attacker poisons the routers' view of the AS topology and change their routing table by flooding false LSA instances. To mitigate this attack, the primary security mechanism deployed in OSPFv2/v3 is the "fight-back" mechanism. Since the victim router will also receive the false instance of its own LSA, it immediately verifies the sequence number of this false instance, and generates a newer instance of the LSA with a higher sequence number which cancels out the false one. This mechanism can mitigate the effects of the attack and prevent most OSPF attacks from persistently falsifying an LSA of another router.

Beside LSA falsification, some security vulnerabilities in OSPFv2/v3 protocol have been found in the past few years. Previous approaches focus on attacks against "fight-back" mechanism [5–7]. Periodic injection, introduced by Jones et al. [8], evades the "fight-back" mechanism by advertising the false LSA within 5 seconds. Nakibly et al. propose Disguised LSA to evade the "fight-back" mechanism [9, 10]. This attack is more severe than periodic injection [8] because it only requires two malicious packets to persistently falsify an LSA of a router. Furthermore, Nakibly et al. identify another serious vulnerability in the most popular OSPF-related routers [11, 12]. An attacker can completely poison of all routers within the AS by sending a false LSA with the same Link State ID of the victim but different routing information. The details of these attacks are given in Section III.

One strong assumption of all mentioned attacks is that the

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

attacker is able to send LSAs to the routers in the routing domain while the routers regard them as valid LSAs. Nakibly et al. think this can be done by an insider, namely an attacker who gains control over a single router in the AS [10, 12]. The attacker can get the ability to control the router by conspiring with an authorized person who has physical access to the router, or by remotely exploiting an implementation vulnerability of the router. Limited by this assumption, these attacks are not practical because the attacker has little chance to control the router in real-world environment. In this paper, we present two novel attacks without having the ability to control the router. These two attacks turn out to be more powerful and simpler than the existing attacks since these attacks can be done by a common host in the AS. We implement a security vulnerability detection system and successfully verify these attacks in both simulations and real-world experiments. These attacks are platform independent. So the malicious program can be implemented on Windows Operating System, which means more harmful than existing attacks.

The paper is organized as follows. Section II gives a brief overview of the OSPF specification and its security mechanism. Section III discusses the previous attack that exploits security vulnerabilities in the design of OSPF specification. Section IV discusses our security analysis and presents the two novel attacks. Section V evaluates the power of these attacks both in simulation environment and real-world environment. Section VI proposes a security vulnerabilities detection system for OSPF protocol to detect the existing vulnerabilities in real-world routers. Finally, Section VII concludes the paper.

II. Background

A. Protocol Fundamentals

Today's Internet uses two classes of dynamic routing protocols: distance-vector and link-state. Distance-vector routing protocol manipulates vectors of distances to other nodes in the networks. Comparing to distance-vector, link-state routing protocol requires routers to inform all nodes when the network topology changes. Every node constructs a map of the network connectivity, in the form of a graph, showing how nodes are connected each other. Each node then independently calculates the least cost paths from it to all possible destinations in the networks. The collection of the least cost paths will then form the node's routing table.

OSPF is one of the most widely used routing protocols on the Internet. It is an interior gateway protocol (IGP) based on link-state technology. It distributes routing information between routers belonging to the same AS. The routing table generation of OSPF is depicted in Fig. 1.

Every OSPF router first sends "hello" packets to discover its neighboring routers. There are three components in the hello packet header to keep information about the status of routers: "hello interval", "router dead interval", and a neighbor list. "hello interval" indicates how frequently the sender should retransmit its hello packets; "router dead interval" tells how long it takes to declare an unavailable router, and the neighbor list describes the neighbors that the sender has already met.

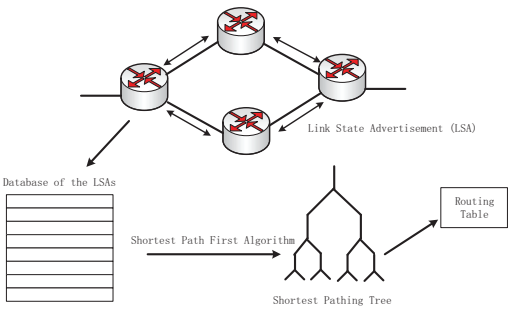

Fig. 1. The routing table generation of OSPF.

Once a neighboring router is found via the hello protocol, the sender goes through a "database exchange process" to synchronize its database. Then the information about the sender's local neighbors is assembled into a LSA, and is broadcasted via a reliable intelligent flooding scheme to all other routers. These LSA packets make up the link state database for the networks. Once all routers have an up-to-date link state database, each router can use the Dijkstra's algorithm to calculate a shortest-path tree with other routers and then form a complete picture of routing in the networks.

The OSPF protocol runs directly over IP, using 89 port. IP reassembling is used when fragmentation is necessary. There are five types of OSPF packets: "hello" packet is used by a router to discover its neighbors; "database description" and "link state request" packets are used to synchronize two routers' databases when an adjacency is being initialized; "link state update" packet is used to update link state database; and "link state acknowledgment" packet ensures a reliable transmission by acknowledging flooded LSAs.

B. security mechanisms

The primary security threat to link-state routing networks is LSA falsification. Several security mechanisms has been discussed in the design of routing protocols to mitigate this attack. As a typical, the security mechanisms deployed in OSPF networks are as follows:

1) Packet authentication: Each OSPF packet should be authenticated in three ways: NULL authentication, simple password authentication, and cryptographic authentication [1]. In these three authentication schemes, only cryptographic authentication can provide integrity protection based on cryptographic algorithm. All routers have a shared secret key attached to a common network/subnet in cryptographic authentication. For each OSPF protocol packet, this key is used to generate/verify a message digest with MD5 hash function. Due to the lack of defined secret key management mechanism, a network operator must manually configure the secrets at every router [13]. Therefore, for many today's ASs, the secret is the same for all their links. Since the security of the MD5 hash function can be easily compromised by dictionary attack and collision attack, the attacker can easily get the secrets of the whole OSPF networks.

2) "Fight-back" mechanism: The OSPF protocol applies a security mechanism, named as "Fight-back" mechanism [14], to detect and cancel the false LSAs flooded in the networks based on its fault-tolerance design. Once a router receives an instance of its own LSA but newer than the last instance it originated, the router immediately advertises a newer instance of the LSA which cancels out the false one. The OSPF protocol also has a flooding mechanism which ensures all routers in the networks maintain the same topological database. Therefore, a false LSA advertised by the attacker will be send to the victim router eventually. The "Fight-back" mechanism and the flooding mechanism ensure that all the LSAs in other routers are originated from the valid router.

III. Attacks on OSPF protocol

Previous approaches have pointed out several security vulnerabilities in these security mechanisms. However, one strong assumption of all mentioned attacks is that the attacker is able to send LSAs to the routers in the routing domain while the routers regard them as valid LSAs. In this paper, we identify several novel vulnerabilities that the attacker can still falsify LSAs to the routers even when he is a common user inside the OSPF networks, and propose two novel attacks: adjacency spoofing attack and single path injection attack. In large service provider and enterprise networks scenarios, the gateway of a subnet often act as an OSPF node in backbone networks. We test the gateways in several campus networks with our security vulnerabilities detection system, and find that almost all gateways act as OSPF nodes in the campus networks and nearly half of them broadcast "Hello" packet in "passive disabled state". Therefore, the host in these subnets can capture the "hello" packets from the gateway and get some useful network parameters of the adjacent routers. With these network parameters, an attack can construct the false LSAs to spoof the routing tables of the whole OSPF networks.

A. Adjacency Spoofing Attack

The gateway dynamically discovers its neighbors by periodically broadcasting "hello" packets on its attached links when it acts as an OSPF router. This packet includes the identities of all the routers. Once the attacker connects to the gateway in the same subnet, he first captures the "hello" packets broadcasted by the gateway and gets the network parameters, such as Router ID, Area ID, HelloInterval, RouterDeadInterval, AuType and so on. These parameters can be used to construct the corresponding "hello" packet related. Notice that The Router ID chosen by the attacker should be larger than the gateway's. Second, since the gateway is assumed to be a designated router, it starts to set up an adjacency with the attacker. The attacker sends the forged "hello" packet as if he is a legal adjacent router to make an adjacency relationship with gateway. Third, the gateway sends a Database description (DBD) message when it receives the "hello" packet from the attacker, and the attacker and gateway exchange the summaries of LSAs in their database with DBD messages. Finally, the gateway requests its peer new instances after the LSA exchange. In this way, the gateway receives the false LSAs from the attacker and floods them to the whole OSPF networks. The procedures of the adjacency spoofing attack is depicted in Fig. 2.

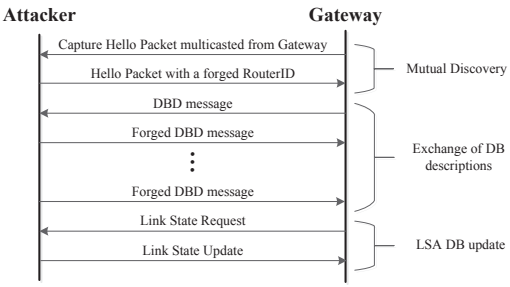

Fig. 2. The procedures of the adjacency spoofing attack.

The victim gateway regards the attacker as a adjacent router after adjacency spoofing attack. The attack can further advertise any false LSA and change any traffic path he wants. All routers will change their routing table after receiving the flooded LSAs. Comparing with the remote false adjacency attack proposed by Nakibly et al. [10], adjacency spoofing attack is more general and powerful, since it is can be used to realize DNS spoofing, phishing Website, eavesdropping, and man-in-the-middle attacks.

B. Single Path Injection Attack

Comparing with "Disguised LSA" attack with two false LSA proposed by Nakibly et al. [10], we find a specific scenario that the attacker only needs to send one false LSA to change the routing table of all routers even considering "fight-back" mechanism.

The original "Disguised LSA" attack exploits a vulnerability in the "Fight-back" mechanism which the victim router does not advertise a correcting LSA if the received LSA is "identical" to its last valid LSA. The OSPF protocol considers two instances of an LSA to be identical if they have the same values in the following three fields: Sequence Number, Checksum, and Age. The key point is that the protocol considers these two LSAs to be the same even when the actual advertised links are different. Disguised LSA attack exploits this vulnerability by advertising a false LSA which seems to be identical to a future fight-back LSA. Therefore, other routers consider the fight-back LSAs as duplicates of the false LSA and ignore them automatically.

We find a new vulnerability in OSPF protocol. Section 13.7 of the OSPF protocol specification defines the process procedures of the received link state acknowledgments: the router ignores this acknowledgment and examines the next acknowledgment when the LSA acknowledge does not have an instance on the link state retransmission list for the neighbor. Otherwise, the router removes the item from the list and examines the next acknowledgment. We exploit this feature to advertise a false LSA via a middle "springboard router" to the specific "polluted router" without triggering the Fight-back mechanism. As depicted in Fig. 3, attacker can send a

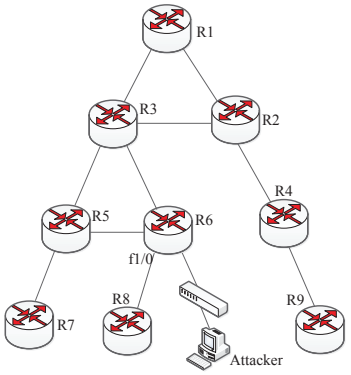

Fig. 3. The network topology in single path injection attack.

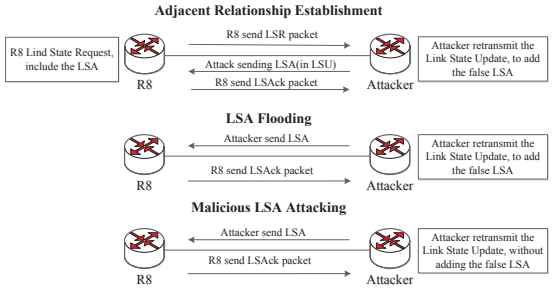

Fig. 4. The interactivities of the link state update process in different steps.

false LSA from the host to the polluted router R8 via the springboard router R6. The source IP address in this false LSA is set to the address of f1/0, and the ID value is set to R6's ID. In this way, the polluted router R8 believes that this false LSA is flooded from the router R6. R8 is the first router to store the false LSA in its link state database and floods the link state acknowledgment to the all other routers in the OSPF networks. By exploiting the OSPF feature mentioned above, attacker can bypass the "Fight-back" mechanism even though the router R6 may receive the link state acknowledgment which includes the false LSA from router R8.

Fig. 4 depicts three situations of sending link state update packet with a LSA. The former two situations display the normal interactive processes in the adjacent relationship establishment stage and the LSA flooding stage. The last one is the interactive process between the attacker and polluted router when sending a false LSA. In the normal stage, a router inserts the LSA into its local link state retransmission list after sending the LSA to other routers, and waits for the link state acknowledgment within a given time interval. If no link state acknowledgment is sent back before the time running out, the router retransmits the LSA. Otherwise, the router confirms the LSA and deletes the LSA from the local link state retransmission list.

When the attacker sends false LSA with the springboard router's ID, the springboard router's link state retransmission list won't contain any information of this false LSA. Therefore, when the springboard router receives a link state acknowledgment which contains the false LSA from the polluted router, it just discards this acknowledgment directly and does not trigger the "Fight-back" mechanism according the feature defined in Section 13.7 of the OSPF protocol.

In procedures of single path injection attack, we can infer that this attack only occurs when there is only one transmission path between the springboard router and polluted router. Therefore, in the topology depicted in Fig. 3, R7 with R5, R8 with R6, R9 with R4, and R4 with R2 meet this the condition of this attack. Once an attacker confirms the springboard router, he constructs a false LSA with the springboard router's ID and sends it to the polluted router. As depicted in Fig. 3, the attacker chooses router R2 as springboard router and router R2 as polluted router, and sends the false LSAs to the router R4. This attack can be further used by black-hole traffic destined attack against a specific subnet by changing the routing table to the specified routers as a black-hole.

IV. Implementation and Experimental Result

In this section we implement a security vulnerability detection system and evaluate the efficiency of the attacks we proposed. The results show that these two attacks are not only practical but also highly effective in both simulation environment and real-world environment.

A. Attack Validation in Simulation Environment

We establish the simulation environment based on GNS3 [15]. GNS3 is a graphic network simulator to design and configure virtual networks. It allows the combination of virtual devices and real devices, and can be used to simulate complex networks. The simulated routers are Cisco routers and their Cisco IOS versions are c3640 by default.

The topology of the network in our simulation is depicted in Fig. 5. The whole simulated OSPF network is divided into three area: the backbone area *Area0*, inter-area *Area1*, and another inter-area *Area2*. The router R9 is a backbone router which connects to a real-world campus network in Southeast University. The IP address of each router is the combination of the router ID and the IP address range of the corresponding subnet. For example, the IP addresses of the two interfaces in router R9 are 30.129.21.9 and 30.129.22.9. C2 and C3 are two internal hosts. We also deploy a real host A2 as an attacker, whose IP address is 192.168.80.2. R3 is A2's adjacent router and its interface f1/0 runs the OSPF protocol. Another attacker A1 is a common host in the subnet 10.129.23.0/24 at *Area1*. We deploy a Vmware virtual machine to act as A1. R10 is A1's adjacent router and its interface f3/0 runs the OSPF protocol. Both A1 and A2 implement the security vulnerabilities detection system with QT for the proposed attacks. The operating systems of the hosts are all windows 7. We also use Wireshark to capture all packets in attacking procedures.

Adjacency Spoofing Attack First A2 runs the detection system to capture the network parameters by sniffing the "hello" packet broadcasted by the adjacent router. The result is depicted in Fig. 6. We can first find that the adjacent router is R3. Then A2 disguise himself as an AS boundary router (ASBR) to establish an adjacent relationship with router R3. The procedures of the adjacent relationship establishment are

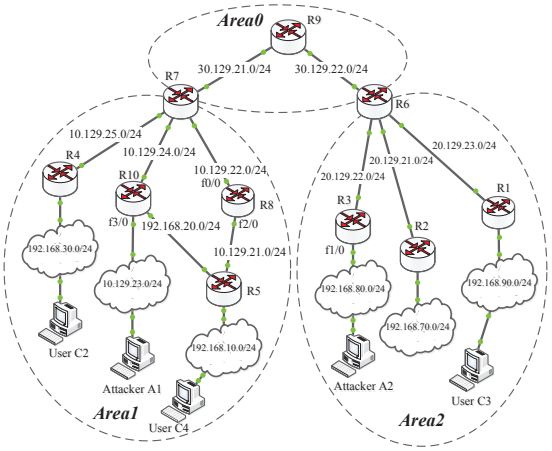

Fig. 5. The network topology of the simulation environment.

Fig. 6. The network parameters of the OSPF networks.

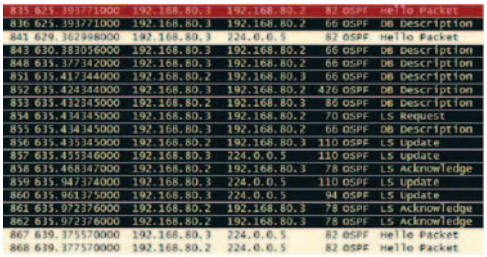

Fig. 7. The procedures of the adjacent relationship establishment.

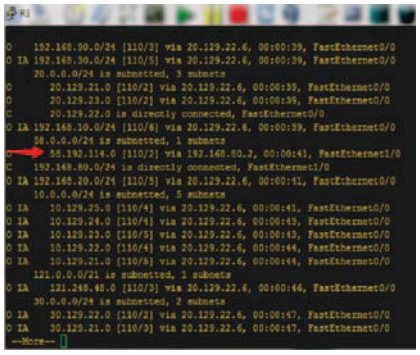

Fig. 8. The routing table of router R3 after the attack.

depicted in Fig. 7, we can find A2 sends several OSPF packets, such as "hello" packet, "DB Description" packets, "LS Update" packets, and "LS Acknowledge" packets. Once A2 becomes the adjacent ASBR of R3, it injects a false LSA with malicious route information of the subnet 58.192.114.0/24. R3 accepts the false LSA and floods it to all other routers. This malicious route information in R3's routing table is depicted in Fig. 8.

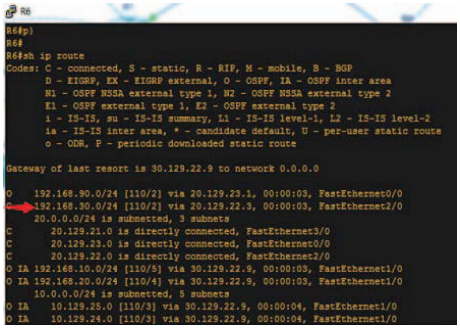

Fig. 9. The routing information of router R6 after the attack.

Single Path Injection Attack In normal condition, the traffic path from User C1 to User C2 is R3-R6-R9-R7-R4. The attacker A2 exploits the single path injection attack to inject a false LSA to route R6. The packets from C1 to C2 are forwarded to the wrong router R2 when passing through R6. Therefore, a traffic black-hole is created. This attack chooses R3 as springboard router and R6 as polluted router. The parameters of the false LSA is depicted in Table I. For this false LSA, the source IP address is set to the springboard router's, and destination IP address is set to the polluted router's. Meanwhile, the router ID in router header of this false LSA is set to 3, as router R3, and the router ID in OSPF header is set to 3.3.3.3, as the identity of router R3. Moreover, the sequence number of this false LSA in LSA header is set to a very large number to ensure that this value is larger than all current LSA sequence number. Finally, the link ID of this false LSA in the router LSA fields is set to 192.168.30.0.

TABLE I. The parameter of a false LSA

Location	Fields	Value and Description
IP header	Source Address	20.129.22.3 (The springboard router R3's interface address)
	Destination Address	20.129.22.6 (The polluted router R6's interface address)
	Protocol Number	89 (OSPF protocol)
	TTL	3
OSPF header	Type	1 (Link State Update Packet)
	Router ID	3.3.3.3 (Use the false identity to disguise as R3)
	Area ID	0.0.0.2 (Area ID)
LSU header	LSA number	1 (the number of link state)
LSA header	Link State Type	1 (Route LSA)
	Link State ID	3.3.3.3
	Announce Router	3.3.3.3 (This is forged route)
	Sequence Number	80000010 (It should be larger than all current SN)
Route LSA	Link ID	192.168.30.0 (Forged subnet)
	Link Data	255.255.255.0 (mask of subnet)
	Link Type	3
	Distance Vector	1
	Link ID	20.129.22.6
	Link Data	20.129.22.3
	Link Type	2
	Distance Vector	1

When the attacker A2 sends the false LSA depicted in Table I, the information of the LSA stored in the router R6 will be changed, as depicted in Fig. 9. The fake network topology showed in router R6 is 192.168.30.0/24, which means all packets transferred from user C3 to user C2 will be forwarded to the router R2 through the router R6. This makes a traffic black-hole to block the traffic between C3 and C2.

Fig. 10. The effect of the route spoofing.

B. Attack Validation in Real-world Environment

We evaluate the security of the OSPF networks in several campus networks of Southeast University and other Universities in Nanjing with our security vulnerabilities detection system. The test result shows that nearly all gateways act as an AS boundary router in OSPF networks and half of them broadcast the "hello" packet which can be easily captured by a common host. By exploiting these vulnerabilities, the attacker can easily launch DNS spoofing, phishing Website, eavesdropping, and man-in-the-middle attacks.

We also evaluate the performance of the proposed attacks in real-world environment with two computers. One acts as the attacker connects to a campus network with wired connection. Another computer acts as a common user accessing to the campus network via Wifi. The IP address of the attacker is 172.21.134.2 in the subnet 172.21.134.0/24, and the IP address of the gateway of this subnet is 172.21.134.1. The common user's IP address is assigned by the Wifi router via DHCP protocols and changes dynamically according his place in campus. We establish a test website with the url *infosec.***.edu.cn* and it's real IP address is 202.***.5.32. We show that the attack can hijack to the common user without intruding the user's computer. The two attack examples in real-world are discussed as follows.

1) Route spoofing: First the attacker establishes a virtual machine with the same IP address as the victim website, 202.***.5.32, and the gateway of this virtual machine is configured to 172.21.134.2. Second, the attacker uses adjacency spoofing attack to make the attacker's computer a boundary router to join the whole OSPF networks. Third, the attacker floods a false LSA to OSPF networks. All the router forwards the url requests of the website *infosec.***.edu.cn* in a campus network to the forged boundary router. Finally, the attacker forwards these url requests to the virtual machine with the same IP address of the real website. The effect of the route spoofing is depicted in Fig. 10. The route spoofing is very dangerous for the forged website since the IP address of forged website is the same as the real website's. The only thing changed in this attack is the route path, which is normally ignored by users. Further attacks, such as password sniffer, DNS spoofing, and phishing website, can be applied based on routing spoof attack.

V. Conclusion

We presented tow novel attacks in OSPF networks: adjacency spoofing attack and single path injection attack. These attacks can be launched by an insider host in the OSPF networks without having the ability to control the router. Simulation and real-world experiment results show that these attacks are more general and powerful than the existing attacks in against OSPF networks. They can efficiently modify the routing tables of routers, and further lead to DNS spoofing, phishing Website, eavesdropping, and man-in-the-middle attacks.

References

[1] M. Gupta and N. Melam, "Authentication/Confidentiality for OSPFv3." RFC 4552, June 2006.

[2] C. Dearlove, T. H. Clausen and U. Herberg, etc., "The Optimized Link State Routing Protocol Version 2." RFC 7181, Oct. 2015.

[3] L. Ginsberg and M. Shand, "Reclassification of RFC 1142 to Historic." RFC 7142, Oct. 2015.

[4] M. Bhatia, S. Hartman and D. Zhang, etc., "Security Extension for OSPFv2 When Using Manual Key Management." RFC 7474, Apr. 2015.

[5] S. Gao, Z. Peng, B. Xiao, A. Hu, and K. Ren, "Flood-defender: Protecting data and control plane resources under sdn-aimed dos attacks," in *Proc. of the IEEE International Conference on Computer Communications (INFOCOM)*, 2017.

[6] S. Goldberg, D. Xiao, E. Tromer, B. Barak, and J. Rexford, "Path-quality monitoring in the presence of adversaries: The secure sketch protocols," *IEEE/ACM Trans. on Networking*, vol. 23, no. 6, pp. 1729–1741, 2015.

[7] R. B. Somanatha and J. W. Atwood, "Router authentication, key management, and adjacency management for securing inter-router control messages," *Computer Networks*, vol. 79, pp. 68–90, 2015.

[8] Jones, Emanuele and Moigne, "OSPF security vulnerabilities analysis," *Work in Progress*, 2006.

[9] K. Alex, G. Dima and N. Gabi, "Owning the Routing Table–New OSPF Attacks," *BlackHat*, 2011.

[10] N. Gabi, K. Alex and G. Dima, etc., "Persistent OSPF Attacks," in *NDSS*, 2012.

[11] S. Adi, G. Orna and N. Gabi, "Finding security vulnerabilities in a network protocol using parameterized systems," in *Computer Aided Verification*, 2013.

[12] N. Gabi, S. Adi and M. Eitan, etc., "OSPF vulnerability to persistent poisoning attacks: a systematic analysis," in *Proc. of the 30th Annual Computer Security Applications Conference*, 2014.

[13] V. Manral, M. Bhatia, and J. Jaeggli, etc., "Issues with existing cryptographic protection methods for routing protocols," tech. rep., 2010.

[14] Wang, Feiyi and Wu, etc., "On the vulnerabilities and protection of OSPF routing protocol," in *Computer Communications and Networks*, IEEE, 1998.

[15] Welsh and Chris, *GNS3 network simulation guide*. Packt Publ., 2013.