



Laboratory II

Binna, Reschenhofer, Schörghofer

Course: Internet Infrastructure and Security

Lecturer: FH-Prof. DI Mag. Dr. Dominik Engel

23.11.2016

Table of Contents

List of Abbreviations	1
1 Topology	2
2 Router Spoofing	4
3 Router Authentication	6
3.1 Plain Text Authentication	6
3.2 MD5 Authentication	8
4 Sniffing Attack and Clear Text Password Router Authentication	10
4.1 Cracking MD5 with "John the Ripper"	10

List of Abbreviations

MOTD	Message of the Day
DCE	Data Communication Endpoint
OSPF	Open Shortest Path First
IP	Internet Protocol
LSR	Link State Routing
AS	Autonomous System
TTL	Time To Live

1 Topology

The following topology (figure 1.1, taken from Moodle instructions) had to be recreated in the lab.

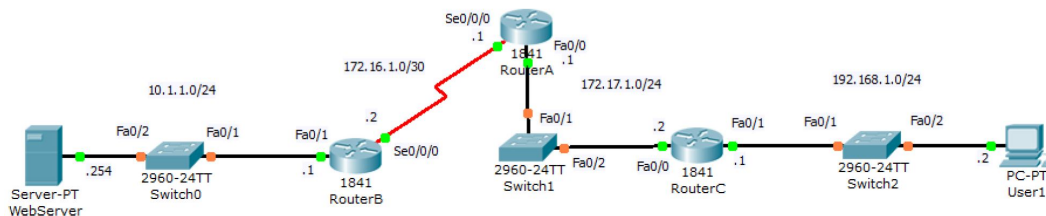


Figure 1.1: Topology (Moodle)

Each device was configured with basic settings like hostname, Message of the Day (MOTD), a secret enable password, disabled IP domain lookup, synchronous logging on line console 0 and the service password-encryption, which prevents passwords from being displayed in clear-text in the start-up and running configuration. The important part is that the clock-rate for the serial interface has to be configured on the Data Communication Endpoint (DCE) device.

```

1 interface Serial0/0/0
2   bandwidth 64
3   ip address 172.16.1.1 255.255.255.252
4   ip ospf authentication message-digest
5   ip ospf message-digest-key 1 md5 7 094F471A1A0A
6   clock rate 64000

```

Listing 1.1: Setting the clock-rate on Router A

Open Shortest Path First (OSPF) is a routing protocol for Internet Protocol (IP) networks. It uses a Link State Routing (LSR) algorithm and falls into the group of interior routing protocols, operating within a single Autonomous System (AS). Concerning the OSPF routing, every router assigns its known networks to the OSPF routing process with area 0.

```
1 router ospf 1
2   area 0 authentication message-digest
3   network 172.16.1.0 0.0.0.3 area 0
4   network 172.17.1.0 0.0.0.255 area 0
```

Listing 1.2: OSPF routing example router A

‘nginx’ has been set up as a webserver on a Linux-PC. The default ‘index.html’ was adapted to show the message "Guter Webserver!" when accessing the server via a web browser.

The IP address has been assigned to ‘10.1.1.254’. Ping and webserver access from the User-PC to the server has been successful as shown in (figure 1.2). The firewall on the Windows machines was disabled to prevent accidental ICMP failures.



Figure 1.2: Webserver access

For this topology the ping command from the user PC to the Linux-PC hosting the webserver took on average of 18ms and had a remaining Time To Live (TTL) value of 61. The ‘tracert’ output as shown in (figure 1.3) from the user PC to the server shows that it takes 4 hops to reach the server.

```
Routenverfolgung zu 10.1.1.254 über maximal 30 Hops

 1   <1 ms   <1 ms   <1 ms   192.168.1.1
 2   <1 ms   <1 ms   <1 ms   172.17.1.1
 3   22 ms   21 ms   21 ms   172.16.1.2
 4   26 ms   26 ms   26 ms   10.1.1.254

Ablaufverfolgung beendet.
```

Figure 1.3: Tracert from user pc to good webserver

2 Router Spoofing

For the router spoofing attack another router has been added to the network as seen in (figure 2.1, taken from Moodle instructions). This router also operates in OSPF mode.

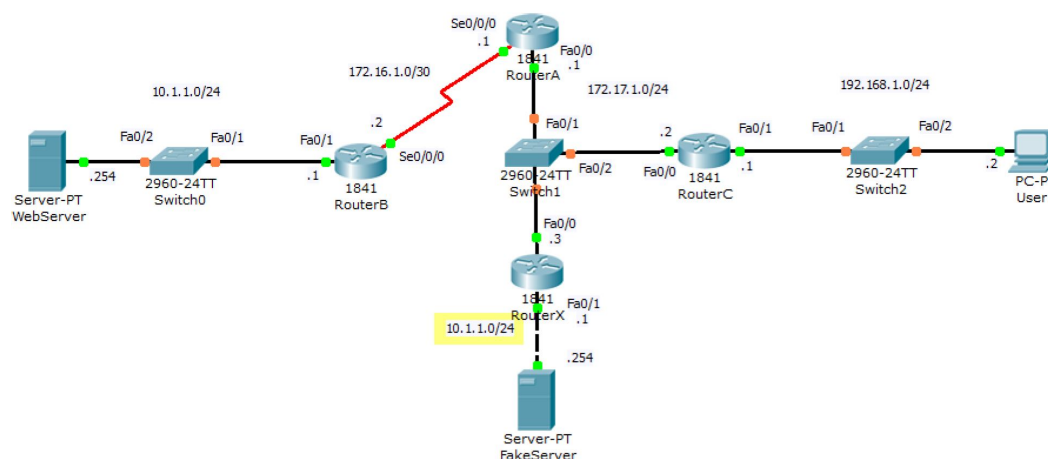


Figure 2.1: Router spoofing topology

Router X is now configured to announce the same networks as Router B. The difference between those routers is that Router X is connected via a faster connection (100 Mbps vs. 64 Kbps) then Router B.

To also see a difference on the User PC, a second 'nginx' was configured on another Linux-PC, this time with "Bad Webserver!" in the 'index.html'.

```

1 router ospf 1
2   network 10.1.1.0 0.0.0.255 area 0
3   network 172.17.1.0 0.0.0.255 area 0

```

Listing 2.1: OSPF configuration on router X

OSPF detects changes in the topology and computes the shortest-path tree for each route using a method based "Dijkstra's algorithm". The bad Router X propagates that he has a better route to network "10.1.1.0/24".

This results in a shorter path from the user pc to the webserver. The user pc doesn't know that this is the bad webserver since OSPF only detected a shorter path to the webserver. The router does not know that the two networks, although having the same subnet, are not the same.

Therefore, an ICMP Echo request (sent with ping) from the user pc will now be routed to the bad server. The latency from the user pc to the bad server is now <1ms with a TTL value of 62. Because of the new route the packet isn't sent over the serial link, instead it is being transmitted via the faster Ethernet connection to Router X. The 'tracert' output (figure 2.2) shows that the packet only needs 3 hops to its destination and has an average response time of <1ms.

```
Routenverfolgung zu 10.1.1.254 über maximal 30 Hops

 1    <1 ms    <1 ms    <1 ms    192.168.1.1
 2    <1 ms    <1 ms    <1 ms    172.17.1.3
 3    <1 ms    <1 ms    <1 ms    10.1.1.254

Ablaufverfolgung beendet.
```

Figure 2.2: Tracert from user pc to bad webserver

If the user accesses the webserver again, he will be redirected to the bad webserver, because of the routing table entry in Router C which states that the shortest path to the network "10.1.1.0/24" is reached via Router X. figure 2.3 shows the output when accessing the webserver again.



Figure 2.3: Bad webserver access

3 Router Authentication

In this chapter, router authentication methods are explained. Router authentication for OSPF allows to flexibly authenticate OSPF neighbours. This enables OSPF routing to exchange routing update information in a secure manner. There are three different types of authentication in OSPF:

- Null authentication: Also called type 0 -> no authentication information is included in the packet header. This is the default setting.
- Plain text authentication: type 1 -> use of simple plain-text passwords
- MD5 authentication: type 2 -> use of MD5 hashed passwords (also not secure anymore!)

OSPFv3 is capable of using SHA1 authentication, which is at least better than MD5 authentication.

3.1 Plain Text Authentication

At first, plain text authentication has been configured. The authentication key is configured for each interface separately by use of the following command:

```
ip ospf authentication-key cisco
```

Only interfaces, where the authentication key match, can participate in OSPF advertising. The following command then enables OSPF authentication for all interfaces inside area 0:

```
router ospf 1  
    area 0 authentication
```

When the commands above are executed solely on Router A and Router C, all OSPF routes that were advertised from Router B are now lost, because Router B doesn't have the authentication configuration.

The main disadvantage of plain text authentication is, that the password can be clearly seen in the packet, as can be seen in figure 3.1.

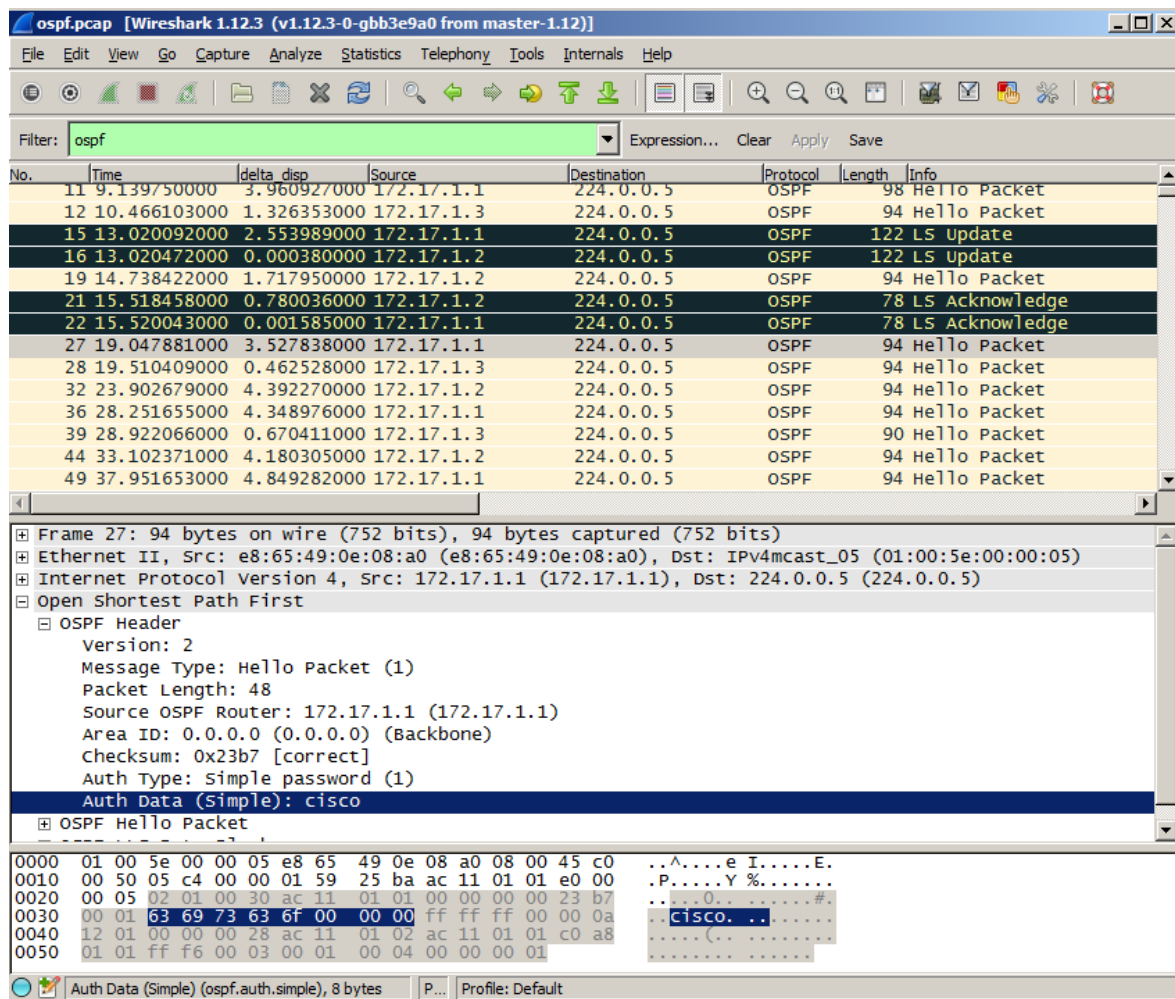


Figure 3.1: Wireshark showing the plaintext password

3.2 MD5 Authentication

First the MD5 message digest has to be set on the interfaces:

```
Interface s0/0/0
    ip ospf message-digest-key 1 md5 cisco
```

After that, MD5 authentication can either be enabled on a per interface basis with the following command:

```
Interface s0/0/0
    ip ospf authentication message-digest
```

Or globally for all interfaces belonging to a specific area:

```
router ospf 1
    area 0 authentication message-digest
```

With activated MD5 authentication, the password is no longer visible in plaintext in the Wireshark capture, as can be seen in figure 3.2.

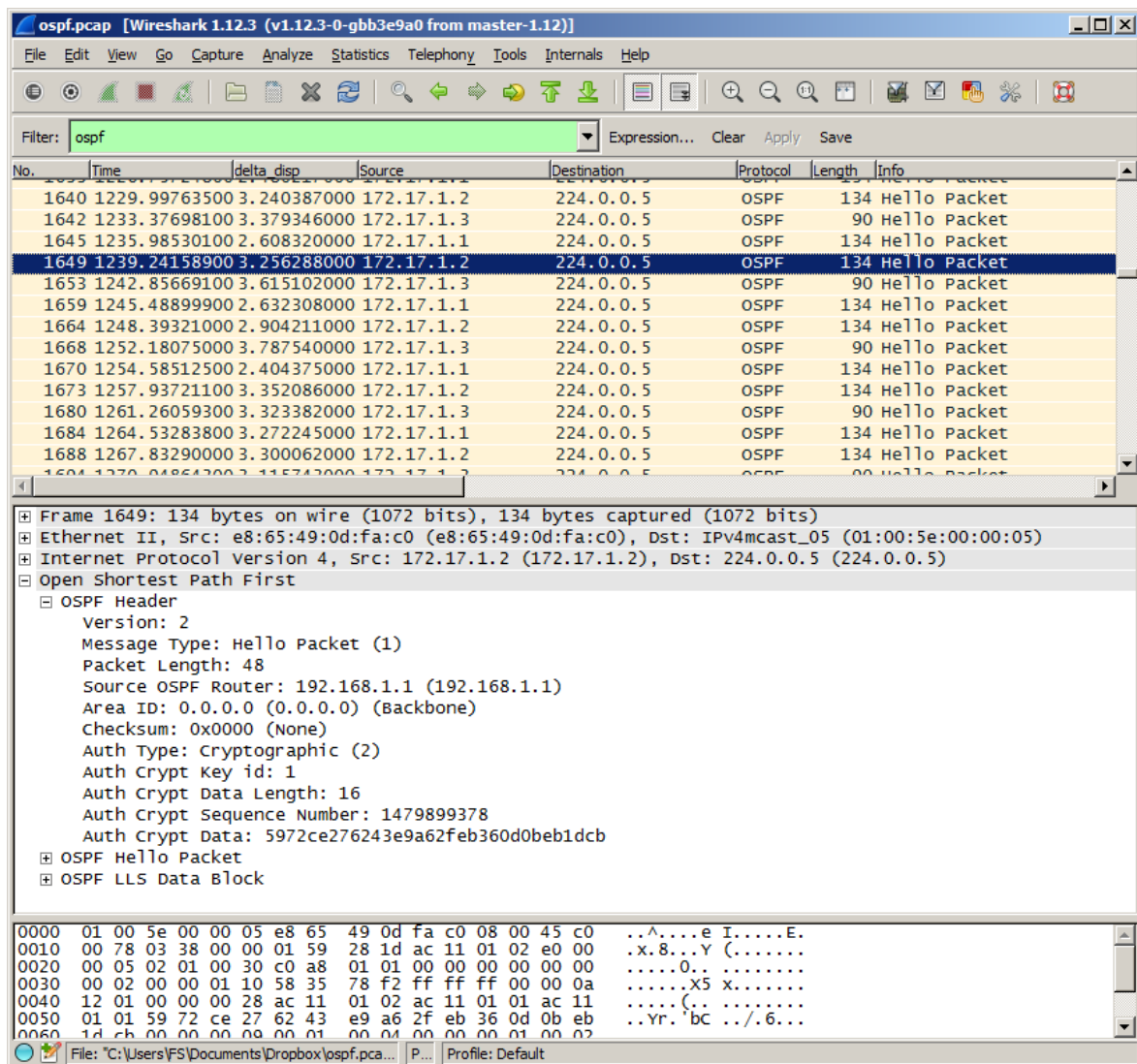


Figure 3.2: OSPF capture with MD5 authentication

4 Sniffing Attack and Clear Text Password Router Authentication

Using a PC connected to Switch1 we were able to capture the OSPF packets seen in the previous section.

The null and plaintext authentication have obvious security problems. An attacker how can sniff packets can then enter the key on the malicious router and announce different routes. Being able to capture the OSPF packets also means the attacker can identify which networks are announced by which router, which allows for easier configuration of the 'bad Webserver'.

Using MD5 as authentication does not reveal the password anymore. But as MD5 can be broken with standard PC hardware in a short amount of time, this is also no longer secure. To avoid replay attacks a sequence number is also introduced. The MD5 hash seen in figure 3.2 is generated from all fields in the packet, except the "Auth Crypt Data" field. This ensures that the sequence number is part of the MD5 hash.

4.1 Cracking MD5 with "John the Ripper"

"John the Ripper" (<http://www.openwall.com/john/>) is a program for cracking different hashing algorithms. We used it to try and calculate the cleartext password from the captured OSPF packets.

To do this, we saved a separate PCAP file which only contained OSPF packets with MD5 authentication.

This file then had to be further processed with 'ettercap' to produce a textfile.