Al-Najah National University
Intrusion Detection System

Exploit JRMI & Analysis

Instructor:Dr.Othman Othman

Mohammed Adnan
Dima Bshara
Sineen Bazyan

1 Producer

1.1 Capture The Packet

1. This module takes advantage of the default configuration of the RMI Registry and RMI Activation services, which allow loading classes from any remote (HTTP) URL. As it invokes a method in the RMI Distributed Garbage Collector which is available via every RMI endpoint, it can be used against both rmiregistry and RMIID, and most other (custom) RMI endpoints as well.
2. So, we Capture the first packet from the attacker server which does RMIloader using **RMI** protocol on the client-side, then it will send another RMILoader to trigger the garbage collector vulnerability to send another RMIPayload, those two packets using the **HTTP** protocol.
3. Figures 1 and 2 shows those captured packet.

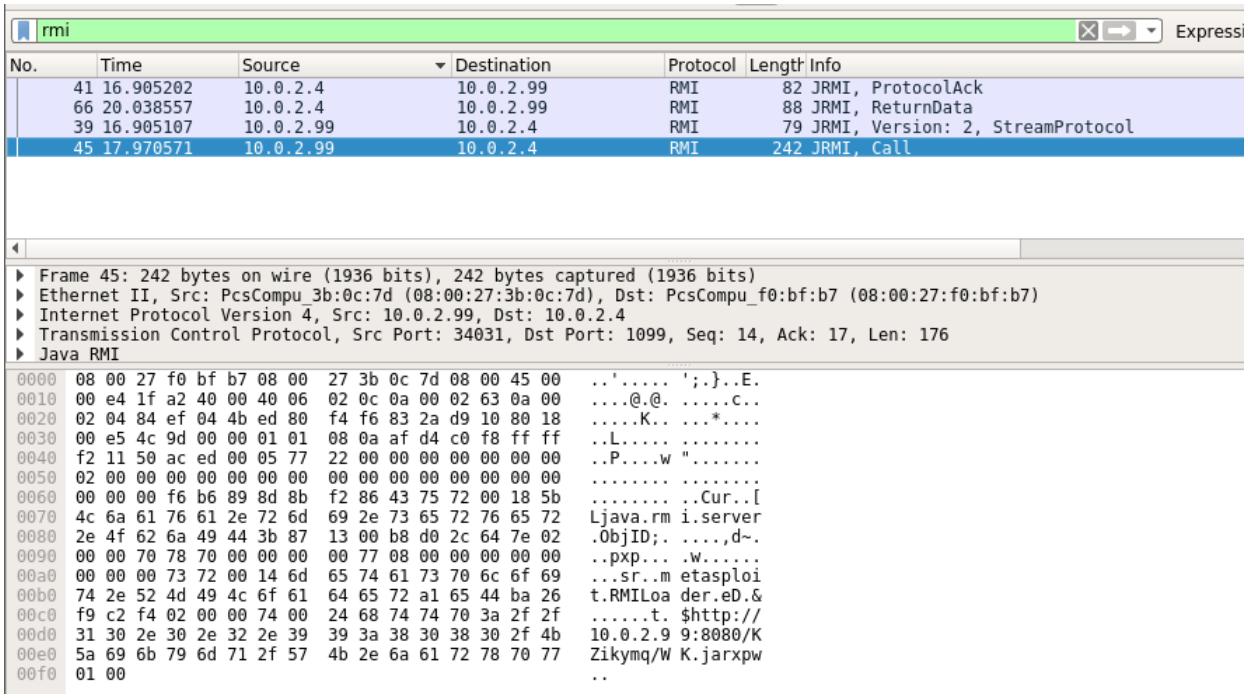

Figure 1: RMILOADER using RMI protocol

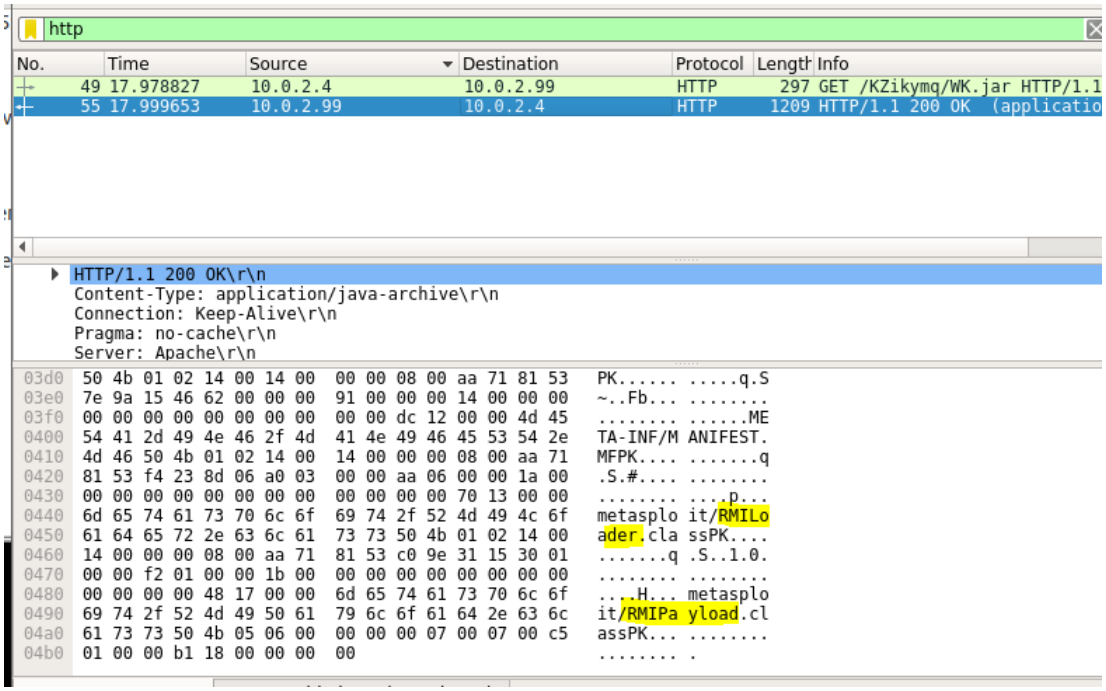

Figure 2: RMILOADER & RMIPAYLOADER using RMI protocol

1.2 Write & test Rule

1. We write the rule-based in our captured, and we do that if the first packet gets loader the second rule will activate dynamically as shown in figure 3.

```
# additions here.

activate tcp any any -> any any (msg:"alert!!"; content:"metasploit";content:".jar";nocase;sid:10000001;rev:1;activates:1;)

activate tcp any any -> any any (activates:1;msg:"alert!! http is loder class" ;content:"RMILoader";content:"RMIPayload";nocase;sid:10000002;)
```

Figure 3: Writing the rule

2. Then, we test the rule as shown in figure 5 to make sure that our rule is working fine.

```
test@test-VirtualBox:/etc/snort/rules$ sudo snort -c local.rules -q -A cmg -r ~/rmi.pcap
12/01-21:13:11.320647  [**] [1:10000001:1] alert!! [**] [Priority: 0] {TCP} 10.0.2.99:34031 -> 10.0.2.4:1099
12/01-21:13:11.320647 08:00:27:3B:0C:7D -> 08:00:27:F0:BF:B7 type:0x800 len:0xF2
10.0.2.99:34031 -> 10.0.2.4:1099 TCP TTL:64 TOS:0x0 ID:8098 IpLen:20 DgmLen:228 DF
***AP*** Seq: 0xED80F4F6  Ack: 0x832AD910  Win: 0xE5  TcpLen: 32
TCP Options (3) => NOP NOP TS: 2949955832 4294963729
50 AC ED 00 05 77 22 00 00 00 00 00 00 00 02 00  P....w".........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00 F6 B6 89 8D 8B F2 86 43 75 72 00 18 5B 4C 6A  ........Cur..[Lj
61 76 61 2E 72 6D 69 2E 73 65 72 76 65 72 2E 4F  ava.rmi.server.O
62 6A 49 44 3B 87 13 00 B8 D0 2C 64 7E 02 00 00  bjID;.....,d~...
70 78 70 00 00 00 00 77 08 00 00 00 00 00 00 00  pxp....w........
00 73 72 00 14 6D 65 74 61 73 70 6C 6F 69 74 2E  .sr..metasploit.
52 4D 49 4C 6F 61 64 65 72 A1 65 44 BA 26 F9 C2  RMILoader.eD.&..
F4 02 00 00 74 00 24 68 74 74 70 3A 2F 2F 31 30  ....t.$http://10
2E 30 2E 32 2E 39 39 3A 38 30 38 30 2F 4B 5A 69  .0.2.99:8080/KZi
6B 79 6D 71 2F 57 4B 2E 6A 61 72 78 70 77 01 00  kymq/WK.jarxpw..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+


12/01-21:13:11.349729  [**] [1:10000002:0] alert!! http is loder class [**] [Priority: 0] {TCP} 10.0.2.99:8080 -> 10.0.2.4:4
12/01-21:13:11.349729 08:00:27:3B:0C:7D -> 08:00:27:F0:BF:B7 type:0x800 len:0x4B9
10.0.2.99:8080 -> 10.0.2.4:43337 TCP TTL:64 TOS:0x0 ID:63027 IpLen:20 DgmLen:1195 DF
***AP*** Seq: 0x1D5B314D  Ack: 0x8410DABC  Win: 0xEB  TcpLen: 32
TCP Options (3) => NOP NOP TS: 2949955862 4294963838
A7 E0 58 56 8F 71 DD 18 33 75 B1 63 DA A2 D0 BB  ..XV.q..3u.c....
18 C5 63 CA 93 E7 2A 3D 9F B5 B9 D1 9B DF 0D 45  ..c...*=.......E
F0 9D 9C D8 A2 5C B6 B8 5F B4 3D DF B0 25 9B 63  .....\.._.=..%.c
```

Figure 4: test the rule

1.3 Insuring that Loader is closed

Figure 5: Resolved

2 References

1- https://knowledgebase.progress.com/articles/Article/How-to-prevent-Java-RMI-class-loader-exploit-with-AdminServer

2- https://github.com/rapid7/metasploit-framework/blob/master//modules/exploits/multi/misc/java$_r mi_s erver.rb$