```
19/12/06 05:20:05 BGP: %ADJCHANGE: neighbor 192.16
 ip route
ernel route, C - connected, S - static, R - RIP,
```

>Analysis of binary protocols – primer on BGP route injection

```
0 via 10.0.2.2, eth2
/24 is directly connected, eth2
0/8 is directly connected, lo
46.0/24 is directly connected, eth4
56.0/24 is directly connected, eth5
192.0/24 [200/0] via 192.168.56.104, eth5, 00:00:
```

Hack In Paris , Jul 2020

Ivica Stipovic

Biography

1. Name>Ivica [Eeveetsa] Stipovic
2. Work>Ward Solutions, Dublin, Ireland
3. Job>Information Security Consultant
4. Contact> Ivica.Stipovic@ward.ie
5. >EOF

Agenda

1. Binary vs Text protocols
2. BGP in a nutshell
3. Understanding BGP session dialog logic
4. Attacking BGP authentication

    -authentication attack

    -route injection attack

1. Limitations of the attack
2. Final Thoughts
3. Demo
4. Q&A

What is binary and what text protocol?

- Quote from Wikepedia: https://en.wikipedia.org/wiki/Binary_protocol

"A **binary protocol** is a protocol which is intended to be read by a machine rather than a human being, as opposed to a plain text protocol such as IRC, SMTP, or HTTP/1.1. Binary protocols have the advantage of terseness, which translates into speed of transmission and interpretation. "

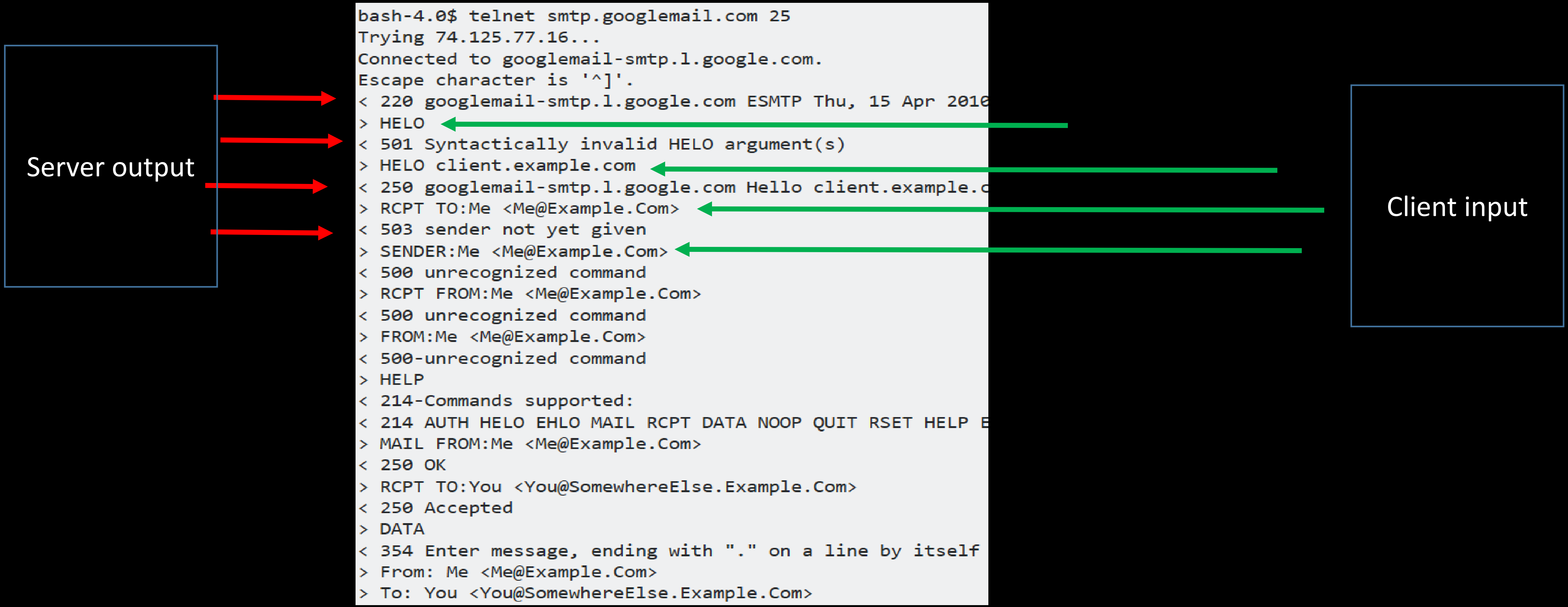
What is binary and what text protocol?

- Examples of text protocols: SMTP, POP3, IMAP, telnet, HTTP, SIP (Session Initiation Protocol (Voice over IP), IRC…)

- Examples of binary protocols: SMB (Server Messge Block/Windows), BGP, RDP (Remote Desktop Protocol/Windows) etc.

What is binary and what text protocol?

- What does it mean “text” protocol? It means that the interaction is text-based (I.E –an SMTP session client input (green) ,server response (red)




```
bash-4.0$ telnet smtp.googlemail.com 25
Trying 74.125.77.16...
Connected to googlemail-smtp.l.google.com.
Escape character is '^]'.
< 220 googlemail-smtp.l.google.com ESMTP Thu, 15 Apr 2010
> HELO
< 501 Syntactically invalid HELO argument(s)
> HELO client.example.com
< 250 googlemail-smtp.l.google.com Hello client.example.c
> RCPT TO:Me <Me@Example.Com>
< 503 sender not yet given
> SENDER:Me <Me@Example.Com>
< 500 unrecognized command
> RCPT FROM:Me <Me@Example.Com>
< 500 unrecognized command
> FROM:Me <Me@Example.Com>
< 500-unrecognized command
> HELP
< 214-Commands supported:
< 214 AUTH HELO EHLO MAIL RCPT DATA NOOP QUIT RSET HELP E
> MAIL FROM:Me <Me@Example.Com>
< 250 OK
> RCPT TO:You <You@SomewhereElse.Example.Com>
< 250 Accepted
> DATA
< 354 Enter message, ending with "." on a line by itself
> From: Me <Me@Example.Com>
> To: You <You@SomewhereElse.Example.Com>
```

Difference between text and binary protocols

- Text protocols allow interaction with humans by accepting text-based input and displaying the text-based output
- Binary protocols require understanding of both the syntax and semantic of the protocol to facilitate a dialog session
- You cannot just telnet to a port that runs binary protocol and interact with it by entering commands/text – nothing will happen
- So, how do we tackle communication with binary protocol?

<u>We need to have a client/agent that “speaks” the designated binary protocol</u>

BGP in a nutshell

- BGP – Border Gateway Protocol
- Enables exchange of the routing information among autonomous systems (AS) on the internet
- BGP uses complex rules based on network policies, paths, rules made by an administrator etc. to make routing decisions
- RFC definition of the BGP is given on the https://tools.ietf.org/html/rfc4271

BGP in a nutshell

- Few attributes important to our attacks are:

- **TYPE** - This 1-octet unsigned integer indicates the type code of the message
  - (OPEN, UPDATE,NOTIFICATION, KEEPALIVE)

- **AS** (Autonomous System) This 2-octet unsigned integer indicates the Autonomous System number of the sender

- **NLRI** (Network Layer Reachability Information)

- **BGP Peer** – BGP neighbour router/process

- **Authentication** – BGP can use authentication to secure the communication with only preapproved IP addresses.

  This is defined as the "**Protection of BGP Sessions via the TCP MD5 Signature Option**"

  Defintion is given in the https://tools.ietf.org/html/rfc2385

BGP in a nutshell

- An example of a simple BGP configuration

```
router bgp 7675
 bgp router-id 192.168.46.3
 network 10.0.2.0/24
 network 10.10.10.0/24
 neighbor 1.2.3.4 remote-as 7675
 neighbor 1.2.3.4 password fgfksdf;lkgsdf;gl
;dlfkjgsdf~jksdfl~'kjg
 neighbor 192.168.56.104 remote-as 7675
 neighbor 192.168.56.104 password Ari&total0
 neighbor 192.168.56.110 remote-as 7675
 neighbor 192.168.56.110 password test
!
```

AS – Local Autonomous System ID

Networks defined

AS- Remote Autonomous System ID

BGP peer IP address

BGP peer specific password

Understanding BGP session dialog logic

Wireshark logic of an unauthenticated BGP session

-Note the messages sequence and their order (OPEN->KEEPALIVE->UPDATE)

-not shown is NOTIFICATION which occurs when some issues with BGP session occurs

Time	Source	Destination	Protocol	Length	Info
0.000000	10.1.1.1	10.1.1.2	TCP	74	41634 → 179 [SYN] Seq=0
0.000174	10.1.1.2	10.1.1.1	TCP	54	179 → 41634 [RST, ACK]
0.450966	10.1.1.2	10.1.1.1	TCP	74	34047 → 179 [SYN] Seq=0
0.454562	10.1.1.1	10.1.1.2	TCP	74	179 → 34047 [SYN, ACK]
0.454691	10.1.1.2	10.1.1.1	TCP	66	34047 → 179 [ACK] Seq=1
0.454878	10.1.1.2	10.1.1.1	BGP	119	OPEN Message
0.461261	10.1.1.1	10.1.1.2	TCP	66	179 → 34047 [ACK] Seq=1
0.461891	10.1.1.1	10.1.1.2	BGP	131	OPEN Message
0.462008	10.1.1.2	10.1.1.1	TCP	66	34047 → 179 [ACK] Seq=5
0.465342	10.1.1.1	10.1.1.2	BGP	85	KEEPALIVE Message
0.465450	10.1.1.2	10.1.1.1	TCP	66	34047 → 179 [ACK] Seq=5
1.452422	10.1.1.2	10.1.1.1	BGP	85	KEEPALIVE Message
1.456193	10.1.1.1	10.1.1.2	BGP	85	KEEPALIVE Message
1.456366	10.1.1.2	10.1.1.1	TCP	66	34047 → 179 [ACK] Seq=7
2.452400	10.1.1.2	10.1.1.1	BGP	89	UPDATE Message
2.492201	10.1.1.1	10.1.1.2	TCP	66	179 → 34047 [ACK] Seq=1
2.492354	10.1.1.2	10.1.1.1	BGP	96	UPDATE Message
2.496062	10.1.1.1	10.1.1.2	TCP	66	179 → 34047 [ACK] Seq=1
4.455131	10.1.1.2	10.1.1.1	BGP	114	UPDATE Message
4.458695	10.1.1.1	10.1.1.2	TCP	66	179 → 34047 [ACK] Seq=1
4.458814	10.1.1.2	10.1.1.1	BGP	132	UPDATE Message
4.463074	10.1.1.1	10.1.1.2	TCP	66	179 → 34047 [ACK] Seq=1

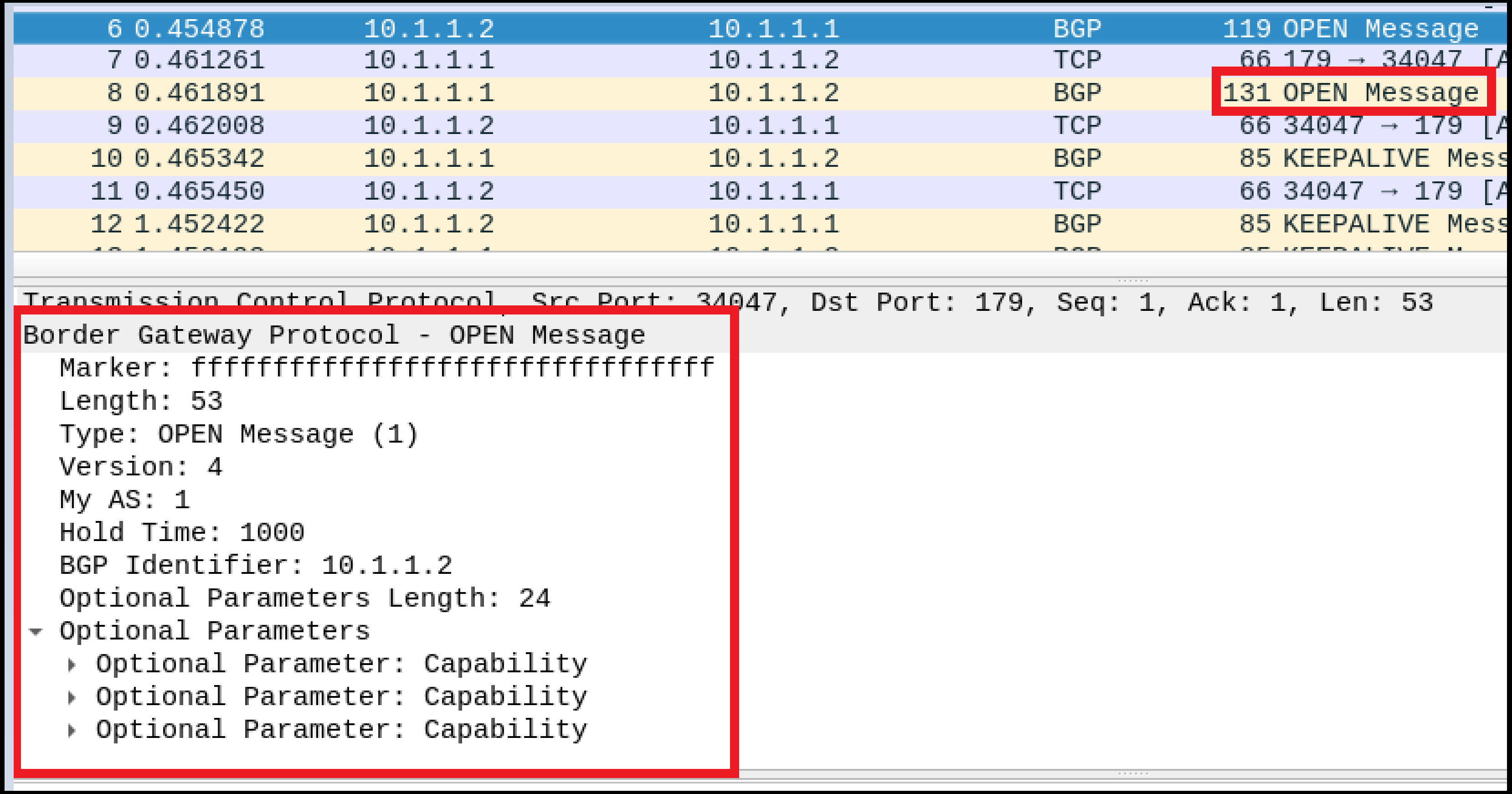
Understanding BGP session dialog logic

Wireshark logic of an unauthenticated BGP session

-Note the BGP attributes of OPEN

-Marker –array of “ff”s , TYPE=OPEN Message

-Version=4 (BGP v4), AS=Autonomous System=1 etc



Understanding BGP session dialog logic

Wireshark logic of an authenticated BGP session

-Note the BGP messages (the same as in unauthenticated – OPEN, KEEPALIVE, UPDATE). We will see the content of “UPDATE” later.

Time	Source	Destination	Protocol	Length	Info
5.375083348	192.168.56.104	192.168.56.101	TCP	74	179 → 55218 [ACK] Seq=1 A
5.375693056	192.168.56.104	192.168.56.101	BGP	119	OPEN Message
5.376308990	192.168.56.101	192.168.56.104	TCP	74	55218 → 179 [ACK] Seq=60
5.376590861	192.168.56.101	192.168.56.104	BGP	93	KEEPALIVE Message
5.378744420	192.168.56.104	192.168.56.101	BGP	93	KEEPALIVE Message
5.379702948	192.168.56.101	192.168.56.104	BGP	93	KEEPALIVE Message
5.379994567	192.168.56.104	192.168.56.101	BGP	97	UPDATE Message
5.420221319	192.168.56.101	192.168.56.104	TCP	74	55218 → 179 [ACK] Seq=98
6.001388631	PcsCompu_a1:8c:8c	Broadcast	ARP	60	Who has 192.168.56.110? T
6.380465437	192.168.56.101	192.168.56.104	BGP	134	UPDATE Message
6.380818829	192.168.56.104	192.168.56.101	BGP	93	KEEPALIVE Message
6.381136942	192.168.56.101	192.168.56.104	TCP	74	55218 → 179 [ACK] Seq=158
6.381159892	192.168.56.104	192.168.56.101	BGP	134	UPDATE Message
6.381527642	192.168.56.101	192.168.56.104	TCP	74	55218 → 179 [ACK] Seq=158
7.001821079	PcsCompu_a1:8c:8c	Broadcast	ARP	60	Who has 192.168.56.110? T

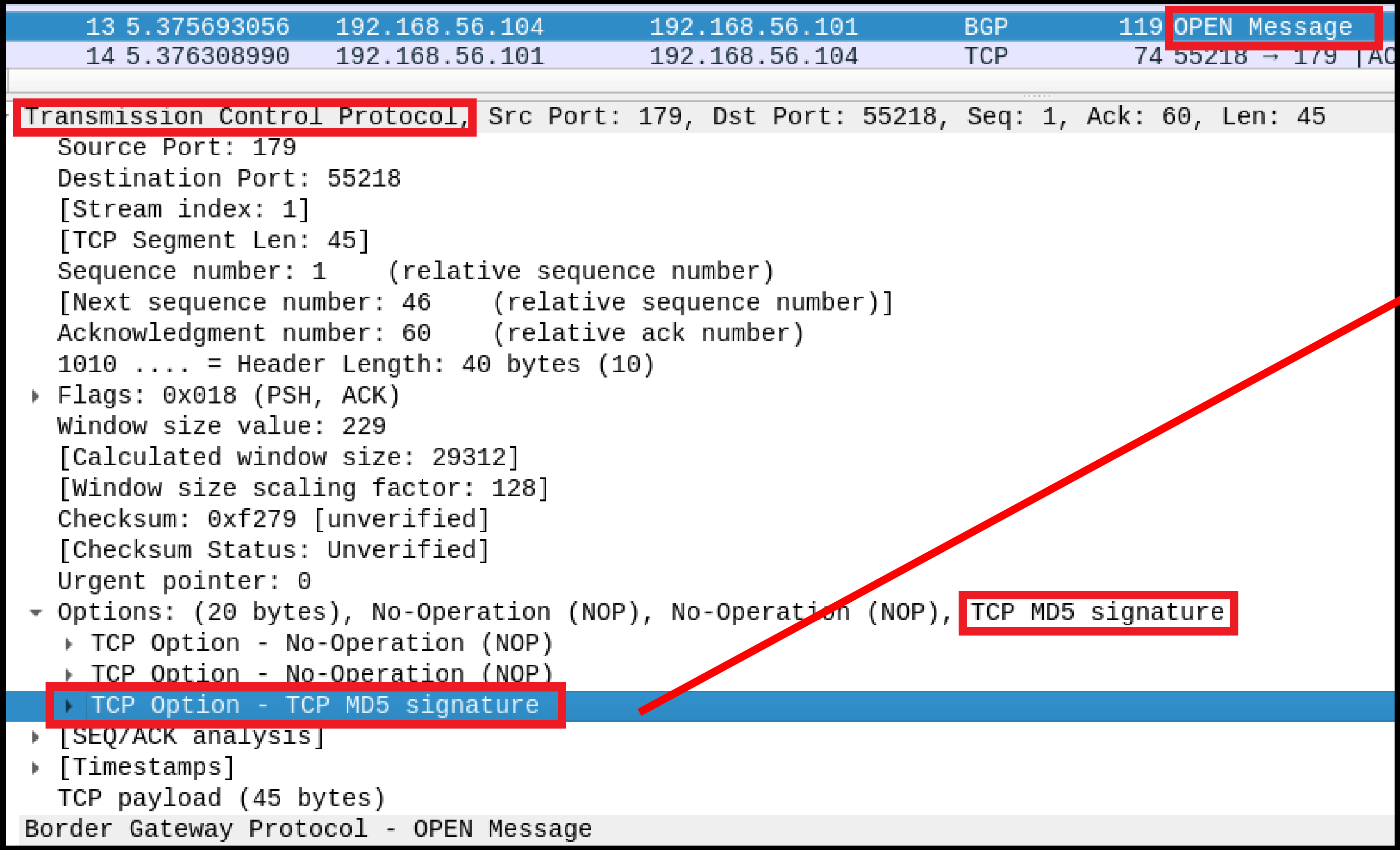
So, where’s the difference to unauthenticated session?

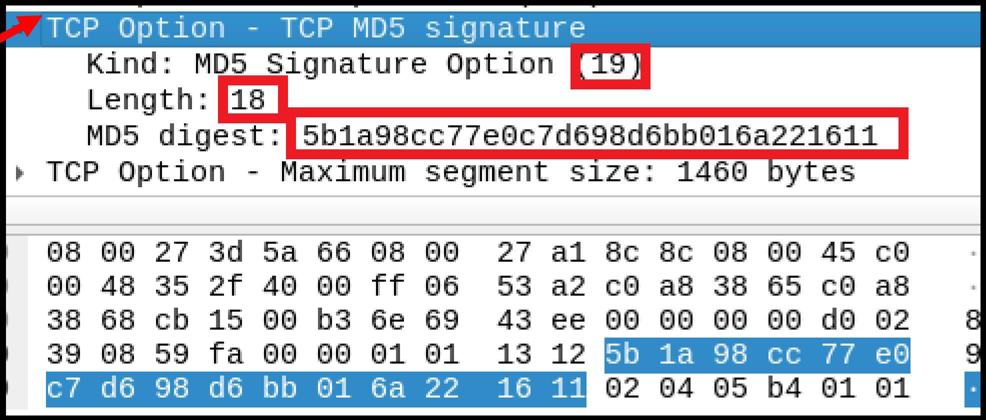
Understanding TCP MD5 signature

Wireshark logic of an authenticated BGP session

-The difference is TCP MD5 signature

-Please note TCP MD5 signature is a part of a TCP header, not the BGP application layer. TCP MD5 signature is what we will be attacking





Understanding TCP MD5 signature

So what is TCP MD5 signature and how is it defined?

RFC2375 (**Protection of BGP Sessions via the TCP MD5 Signature Option**) explains that (https://tools.ietf.org/html/rfc2385)

```
2.0  Proposal

   Every segment sent on a TCP connection to be protected against
   spoofing will contain the 16-byte MD5 digest produced by applying the
   MD5 algorithm to these items in the following order:

      1. the TCP pseudo-header (in the order: source IP address,
         destination IP address, zero-padded protocol number, and
         segment length)
      2. the TCP header, excluding options, and assuming a checksum of
         zero
      3. the TCP segment data (if any)
      4. an independently-specified key or password, known to both TCPs
         and presumably connection-specific
```

TCP MD5 signature = MD5(pseudoheader+TCP header+TCP data+<our password>)

Understanding TCP MD5 signature

This is how the construct for MD5 signature looks like

TCP MD5 signature = MD5(pseudoheader+TCP header+TCP data+<our password>)

```
//construct for MD5 hash - step 1 is pseudo header
memcpy (paket, (char *)pseudopointer,sizeof(pseudo_header));
..
...

//construct for MD5 hash - step 2 is tcp header excluding OPTIONS
memcpy (paket+sizeof(pseudo_header), (char *)tcp,20);
..
...
....
//construct for MD5 hash - step 3 is TCP segment (if any)
memcpy (paket+sizeof(pseudo_header)+20, (char *)tcp+tcp->doff*4,size_of_tcp_data);
//construct for MD5 hash - step 4 is key
memcpy (paket+sizeof(pseudo_header)+20+size_of_tcp_data,key,strlen(key)-1);

MD5(paket, sizeof(pseudo_header)+20+size_of_tcp_data+strlen(key)-1,md5_digest2);
```

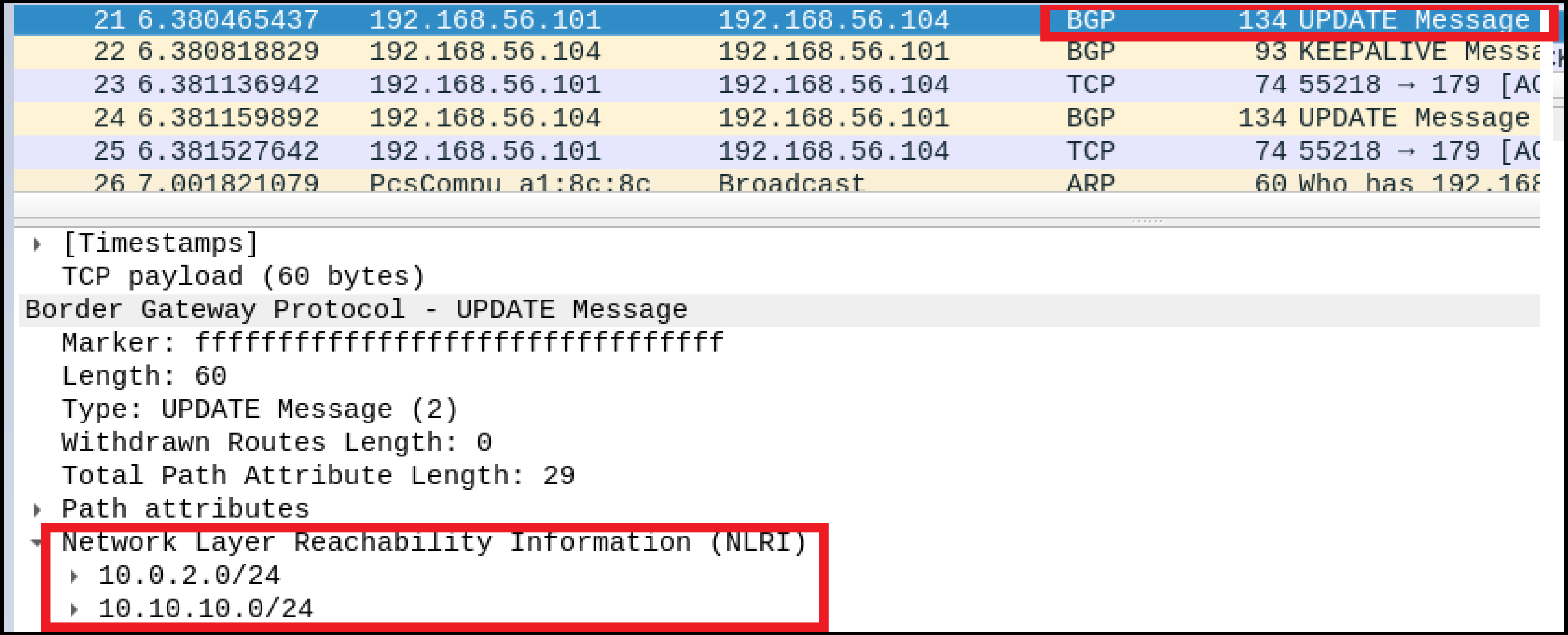
Calculating MD5 hash

Understanding route injection

Almost there, folks, just one more “theoretical” detail

How and where do we inject our route?

In the UPDATE message, more precisely in the NLRI



OK, now we understand the building blocks –let’s attack!

Attacking BGP authentication -anatomy of the attack

Attack is executed in two phases

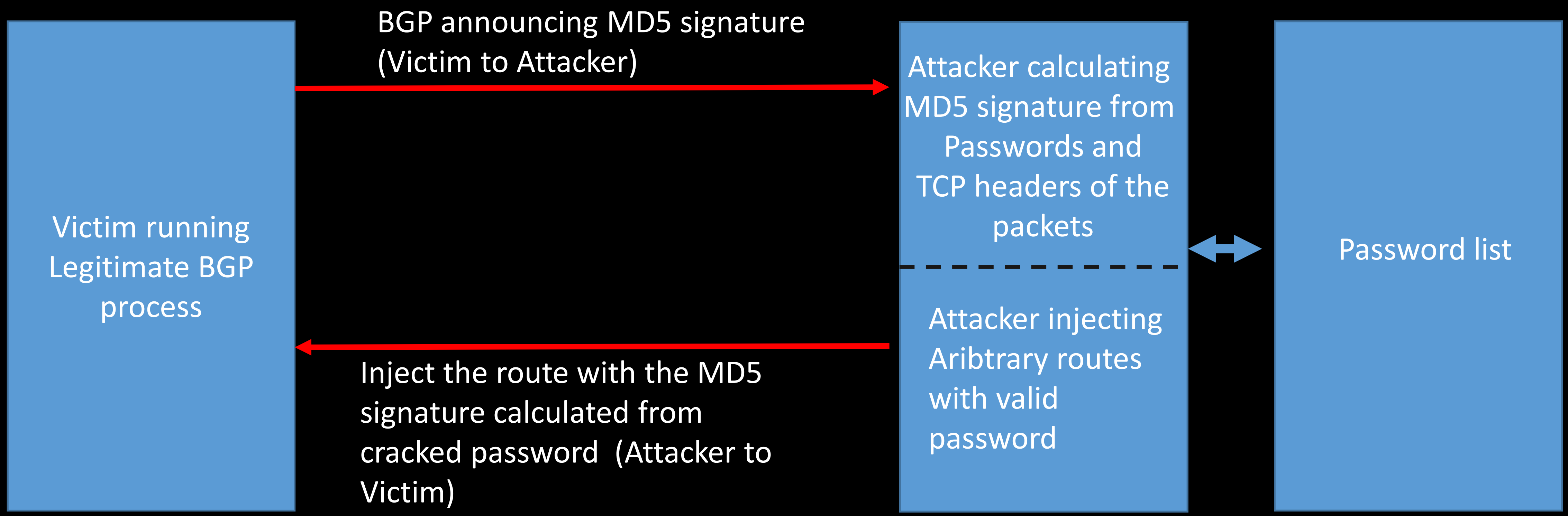
- 1st phase cracks the password so we can authenticate our “intruder BGP process” (attacking TCP MD5 signature)

- 2nd phase will inject arbitrary routes into the legitimate BGP process (UPDATE and NLRI injection)

- There is nothing preventing us from injecting/modifying/deleting virtually any BGP parameter in phase 2, I leave this as an exercise to the audience

- Route injection was selected simply to demonstrate the Proof of Concept

Attacking BGP authentication



Attacking BGP authentication -anatomy of the attack

This is the algorithm for BGP authentication attack -1st phase

```
While (BGP packets are coming)
{
Sniff the SYNs from BGP and locate TCP MD5 signature
If signature found then {
          copy the signature to A
          calculate "my signature" with passwords from text file to B
          if A==B print password
          }
Else print "No signature found"
}
End of password cracking
```

Attacking BGP authentication -anatomy of the attack

This is the algorithm for BGP injection route -2nd stage

Initiate the session with BGP

{

Inject the TCP MD5 signature that we found in step 1

{OPEN} BGP connection

{KEEPALIVE} session

{UPDATE} with our arbitrary route

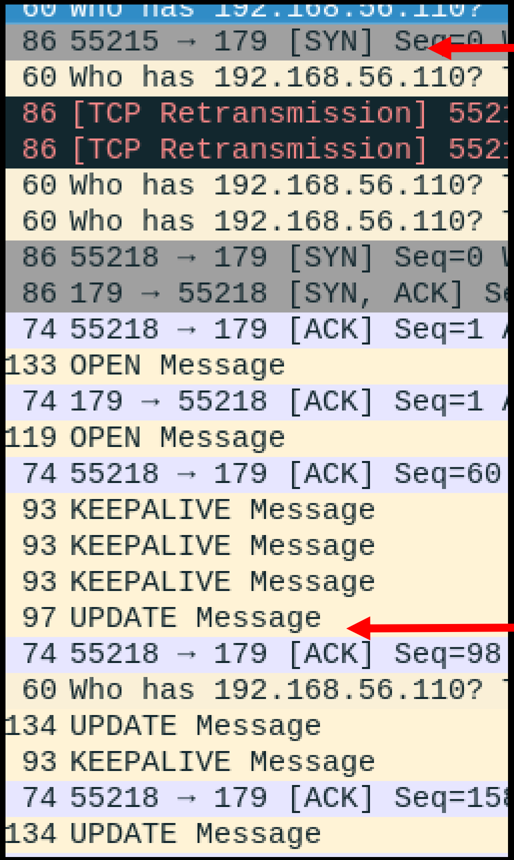
If response came (UPDATE message from BGP) print "injection done"

    else

    print "injection failed" (probably some issue in NOTIFICATION message)

}

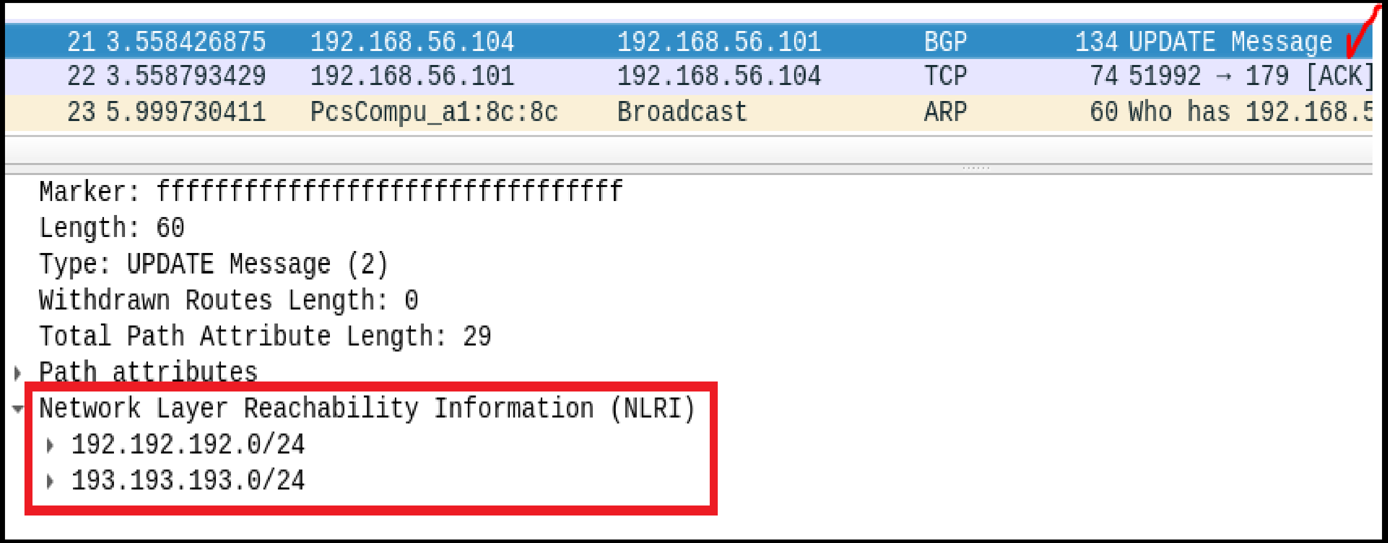
Attack diagram



MD5 signature must be cracked to allow TCP handshake (SYN, SYN+ACK)

After the SYN/SYN+ACT handshake, establish BGP session
-Notice the sequence of OPEN ->KEEPALIVE->UPDATE messages

Route injection happens in the “UPDATE” message, NLRI
NOTE 192.192.192.0/24 and 193.193.193.0/24



Limitations of the attack

-must know the AS (possible to find via internet queries vs AS). Possible (next slide)

-must know authorised BGP peer IP address. Possible by bruteforcing IP within known range-range can be found via internet(next slide)

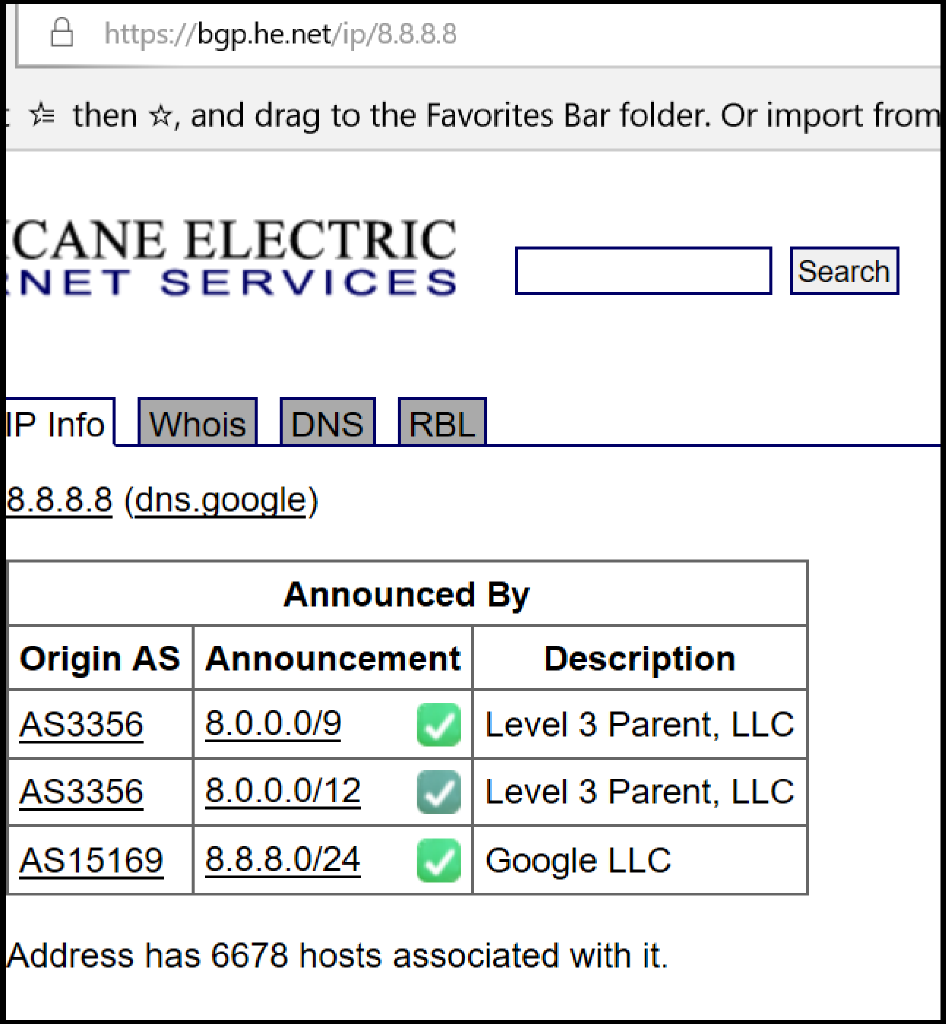
-may require ARP poisoning (difficult - requires access to provieders environment, easy in lab). Difficult.

-limitations of dictionary based password attack (password may not be in dictionary, may last long if list is big). A bit of luck…

-BGP prefix filtering may thwart the attack,TTL (Time-to-Live) limiter or RIPE database ownership query. Up to provider to configure.

Addressing the limitations of the attack

-know the AS (possible to find via internet queries vs AS –enter IP address or AS)


https://bgp.he.net/ip/8.8.8.8

then ☆, and drag to the Favorites Bar folder. Or import from

CANE ELECTRIC
NET SERVICES

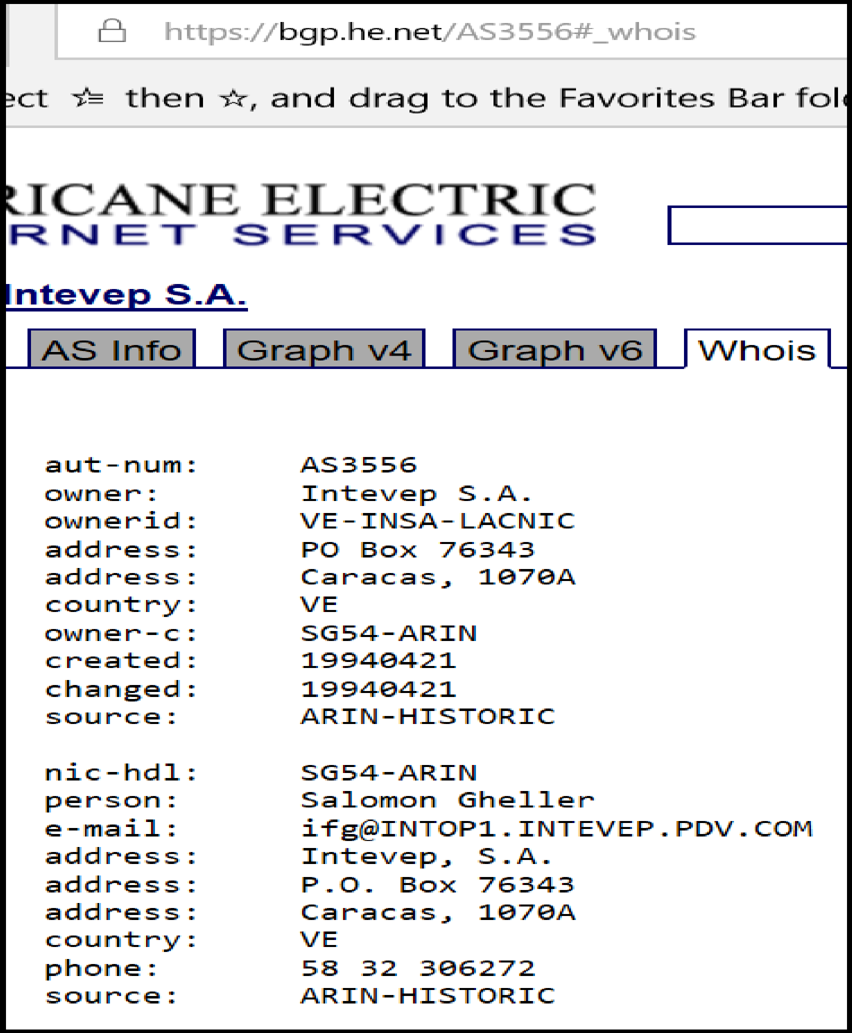
Search

IP Info | Whois | DNS | RBL

8.8.8.8 (dns.google)

Announced By		
Origin AS	**Announcement**	**Description**
AS3356	8.0.0.0/9	Level 3 Parent, LLC
AS3356	8.0.0.0/12	Level 3 Parent, LLC
AS15169	8.8.8.0/24	Google LLC

Address has 6678 hosts associated with it.


https://bgp.he.net/AS3556#_whois

ect then ☆, and drag to the Favorites Bar fol

ICANE ELECTRIC
RNET SERVICES

Intevep S.A.

AS Info | Graph v4 | Graph v6 | Whois

```
aut-num:     AS3556
owner:       Intevep S.A.
ownerid:     VE-INSA-LACNIC
address:     PO Box 76343
address:     Caracas, 1070A
country:     VE
owner-c:     SG54-ARIN
created:     19940421
changed:     19940421
source:      ARIN-HISTORIC

nic-hdl:     SG54-ARIN
person:      Salomon Gheller
e-mail:      ifg@INTOP1.INTEVEP.PDV.COM
address:     Intevep, S.A.
address:     P.O. Box 76343
address:     Caracas, 1070A
country:     VE
phone:       58 32 306272
source:      ARIN-HISTORIC
```

Addressing the limitations of the attack

-know the AS authorised BGP peer IP address – possible to find (at least ranges) from within “Prefixes v4” field

AS Info | Graph v4 | Graph v6 | Prefixes v4 | Prefixes v6 | Peers v4 | Peers v6 | Whois | IRR | IX

Prefix	Description
5.152.177.0/24	HKG-E 5.152.177.0/24
5.152.179.0/24	SJC-C 5.152.179.0/24 SYD-V 5.152.180.0/24
5.152.181.0/24	DAL-A 5.152.181.0/24
5.152.182.0/24	MMI-A 5.152.182.0/24
5.152.183.0/24	MMI-A 5.152.183.0/24
12.177.5.0/24	BLOOMINGDALE HOME TELEPHONE COMPANY, INC
12.192.16.0/24	BLOOMINGDALE HOME TELEPHONE COMPANY, INC
12.192.17.0/24	BLOOMINGDALE HOME TELEPHONE COMPANY, INC
23.142.192.0/24	Kingsburg Media Foundation
23.164.160.0/24	NETWAVE BROADBAND INC
23.175.160.0/24	Pueblo of Santa Ana
27.50.32.0/21	Hurricane Electric (Hong Kong) Ltd

Final thoughts

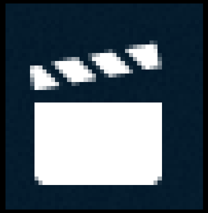
PoC demonstrates that BGP authentication alone is not a bullet proof protection

-The attack shown here works ok in the lab, however, in the real life it would require access/control over the provider’s infrastructure (at least to perform ARP poisoning or some other trick so that the attacker can capture BGP traffic).

-The defenders should deploy not only BGP authentication, but also do the prefix filtering (control of what routes they import), limit TTL (so they know the valid routes are 1 or 2 hops away), they can cross-check the IP address of the route versus RIPE database to ensure the route originates from trusted provider

-Given all that, my proposal for this risk profile is Medium to Low.

Demo



Questions?

```
Telling INIT to go to single user mode.
init: rc main process (2205) killed by TERM signal
[root@centos-4 /]# _
```

Shutting down