BGP MD5:
Good, Bad, Ugly?

Tom Scholl, AT&T Labs

NANOG 39
February 06, 2007

WARNING

BGP MD5 CAN MAKE YOUR ROUTER WEAK

BGP MD5 may cause router impotence due to increased packet flow to the route processor. This can prevent your router from properly functioning.

Health Canada

du MAURIER

MD 5

BGP MD5 – What is it solving?

- A method to authenticate the identity of the remote BGP neighbor.
  - Prevents a session from being hijacked/severed
  - Prevents a misconfiguration from turning up a session with an unauthorized party
- Well-advertised workaround to TCP/IP issue reported two years ago

Background: BGP MD5 (RFC 2385)

- Not a function of BGP
- 18-20 bytes of data in the TCP options field
- Calculated based upon:
  - TCP pseudo-header (src ip, dst ip, ip protocol, segment length)
  - TCP header (excluding options)
  - TCP segment data (if any)
  - The key/password itself

But what about that vulnerability?!?!1111

- One of the issues was that many platforms were not validating ICMP messages such as unreachable messages
  - An attacker could transmit spoofed ICMP error messages to sever a BGP session
- Another issue was that someone could attempt to brute force by transmitting RST's to a wide range of ports and sequence numbers
- MD5 was overwhelmingly suggested as a workaround method to prevent this type of attack without upgrading code
  - People knew something was up when a large portion of the peering community started requesting MD5 on BGP sessions

BGP MD5 is not easy for everyone

- Some organizations provisioning tools required upgrading and human intervention to enable MD5
- The storing of the password in a database or CRM tool presents its own security issues
- How do you securely transmit the clear text password?
- How are you generating the password (using some java interface on some public IT website?)

How to properly exchange MD5 keys
Pig Latin + Cone of Silence

What is the real impact of BGP MD5?

- Examining a MD5 hash in the TCP header adds additional work to a router.
- What if an attacker can spoof with incorrect MD5 hashes to make your router work a bit more?
  - Someone shouldn't be spoofing in the first place if anti-spoofing filters were applied at the perimeter. However...this is not always feasible (think eBGP customers, shared LAN Internet exchanges)
- Perhaps the widespread use of BGP MD5 really is not the only or best solution going forward

What other options are there to protect BGP sessions?

- BGP TTL security check (GTSM)
- BGP session over a separate “protected” interface (dot1q, dlci, tunnel, pseudowire)
- eBGP multihop
- Anti-spoofing ACLs

What was tested?

- Platforms examined:
  - Cisco GSR (GRP-B) running 12.0(27)S
  - Cisco GSR (PRP-1) running 12.0(27)S
  - Cisco GSR (PRP-2) running 12.0(27)S
  - Juniper T320 (RE-2000) running JunOS 8.x
  - Juniper M40e (RE-3.0) running JunOS 8.x
  - Cisco 6500 (Sup720) running 12.2(17)S

What sorts of attacks

- eBGP session with MD5 configured
  - Sending incorrect MD5 hash
  - Sending no MD5
- eBGP session with MD5 configured and GTSM
  - Sending bad MD5
  - Sending no MD5

Testing Results

- Some platforms log BGP MD5 related errors (invalid hash, no hash).
  - This results in additional CPU be spent on logging processes. This can be even worse if you are logging to the console.
- Some platforms that can not do the TTL check in hardware exhibits worse or the same performance hit as without TTL checking enabled.
  - Example: Cisco GSR GRP-B resulted in 50% increase in CPU utilization at low (1,000pps) speed attacks.
- Some platforms examine the MD5 hash before TTL check and log accordingly.

Testing Results (cont'd)

- The actual difference between packets with a signed MD5 hash versus those without was very minimal on most platforms.
- Most modern route processors showed a difference of less than 10% of CPU utilization.
- Using MD5 as an attack vector is useless; packets-per-second (signed or unsigned) will impact you the most.
- The use of TTL checking made no significant difference either (when checked 'in software').

What are the impacts of the alternatives?

- TTL checking is great, if done in hardware (not being performed in the route processor CPU)
  - Cisco CRS-1 can
  - Juniper M120, M320, T320, T640 can
  - All other Cisco devices seem incapable
  - All other Juniper devices seem incapable

(If the above is not accurate, please let me know)

What about alternative methods to peer BGP?

- eBGP Multihop
  - Security through Obscurity
  - No failure detection (yet)

- Separate Interface / Tunnel for BGP traffic
  - Additional complexity
  - BGP next-hop's must be re-written

Looking at the big picture

- The real problem is not MD5 or IP TTL taxing the CPU, the issue is why are those packets touching your route-processor in the first place.

  Perhaps we should try to fix that problem first.

- Can networks mutually agree to not allow their customers to transmit to a peers side of a BGP session?
  - Not easy to do – some egress ACLs permit packets sourced from the route processor, some do not.
  - Do not advertise peer point-to-point interfaces into your IGP/iBGP?

Looking at the big picture (cont'd)

- Should well-known Internet exchange fabric IP address space be treated and protected as your own infrastructure?
  - Drop any packets from customers destined to exchange space.
  - Still does not fix issues with private peering circuits or customer eBGP sessions.
- How about do not advertise IXP space within your IGP?
  - Do not even let your customers route to it in the first place.
  - Probably difficult, since people advertise IXP space in BGP frequently today

So, what can you do?

- Juniper (M120, M320, T320, T640)
  - Ingress ACL that matches on IP TTL value & eBGP source/destination.
- Cisco (CRS-1)
- Other vendors?
- Use caution with route processor policers that may result in you dropping good traffic
  - Software CoPP may actually make things worse

Thank You

Tom Scholl **tom.scholl@att.com**