HopSphere Radius Security

The effectiveness of protecting high value servers through limiting the range of data transmission

Bill Alderson

Chief Technology Officer

bill.alderson@hopzero.com

Ben Haley

SVP Development

ben.haley@hopzero.com

First delivery of the paper:

February 25, 2019

HOPZERO, Inc

11400 Concordia University Drive

Austin, Texas 78726

Abstract

Data security is a growing challenge to governments, businesses, and economies around the globe. Traditional methods, focused on denying network access, have not been successful stopping the ever-looming threat. Hackers continue to create advanced techniques to gain entrance to capture, ransom, exfiltrate and destroy sensitive data. A paradigm shift is needed in how we design data security as the data travel default setting allows global travel – even for devices needing only to communicate inside a datacenter or organization. This paper examines the approach of limiting the distance data is allowed to travel, putting data "on a leash". We will prove that reducing data travel distance results in fewer reachable devices, thereby reducing attack surface. Accessibility to fewer global network devices means fewer hackers can reach, or be reached, by the computer, improving security. Observing and learning data travel patterns enables the determination of hops taken or needed between devices across networks. Enabling communications inside a datacenter for instance, but not allowing access to the Internet is a powerful method to stop hackers. This method, HopSphere Radius Security, is a new advanced technology, intelligently learning, setting, enforcing and alarming when attempts to exceed limits occur. We will prove limiting data travel distance reduces the attack surface, making sensitive servers and databases less vulnerable to attack or exfiltration attempts. Allowing sensitive servers to reach or be reached globally significantly increases risk.

Introduction

HopSphere Radius Security represents a fundamentally new approach, keeping data inside a restricted environment – a powerful solution to secure the most critical data. The system learns the appropriate travel distance to keep data safe inside the datacenter or organization.

The core technology leverages existing standards, used in every TCP/IP device and router since the beginning of the Internet - creating a virtual leash for data. This leash can be used to prevent information from leaving an environment or to limit how far guests can enter an environment. Each operating system starts with enough hop budget to travel the globe unrestricted, often several times over.

After learning the actual travel limits a device or server uses to conduct appropriate communications, a lower setting replaces the global default setting to reduce exposure surface, keeping data inside a smaller “sphere of trust”. This allows the device to conduct business but restricts it from traveling further to dangerous or untrusted locations. Key databases need only communicate inside the data center(s), not directly to internal users or external devices beyond the sphere of trust.

The approach establishes a travel distance budget for each device. As the data packet traverses a router, routers act like a toll taker, decrementing the budget by one. When the travel budget reaches zero, the packet is discarded. Packet travel budget was designed to stop packets from remaining in the network forever due to route loops by decrementing hops as packets traverse. Based on the default actions of a router, we can identify and establish a safer budget to limit how far data may travel.

HopSphere Radius Security makes it impossible to access systems from outside the sphere of trust created by the lower packet lifetime. Attempts to enter the sphere of trust result in a powerful, actionable yet silent, alarm discovering and reporting the hacker intrusion attempt. Because alarms are silent, the hacker has no knowledge of being found out until their access is blocked. This disrupts a hacker’s ability to compromise systems. Data owners immediately know where to deploy resources to resolve the threat, while the data is not compromised.

A. The Problem

Frequent compromises expose the vulnerabilities of firewalls. In response, Intrusion Detection Systems (IDS) were created to discover harmful attachments and viruses slipping through firewalls. Intrusion Prevention Systems (IPS) followed to quarantine and remediate identified threats. Antivirus software, identity access management, and multifactor authentication are attempts to block access to systems after the network has been penetrated. Despite broad application of these technologies, cyber breaches continue to occur at an alarming frequency.

Because firewalls make binary allow/deny decisions for passing data, once an allow decision has been made, the data can travel the remaining distance outside the firewall without limit. Firewalls are improved with the data travel limits applied by HopSphere Radius Security.

Sensitive data might be analogous to a prized pet dog, kept on a leash to control where it goes. If a prized Labradoodle has no leash, it may run into the street, get lost or stolen. When the pet dog is kept on a leash, the owner maintains control. We can use a similar approach with networks. It's time to put sensitive data on a virtual “leash“.

B. The Shift

Security practitioners focus most of their efforts on keeping intruders out. HopSphere Radius Security introduces a radical paradigm shift: focus on keeping communications inside a safe boundary or sphere of trust. This approach creates an entirely new category of enterprise security that puts a virtual leash on sensitive data so that, even if bad actors breach your traditional security systems, they cannot steal your secrets.

The Internet Protocol (the IP of TCP/IP) holds the key to a simple but elegant way for an IP packet to protect itself from cyberattacks. The IP protocol operating inside all computers gives every packet a default lifespan measured in “hops”. Layer 3 switches and routers decrement the hops as data traverses each router.

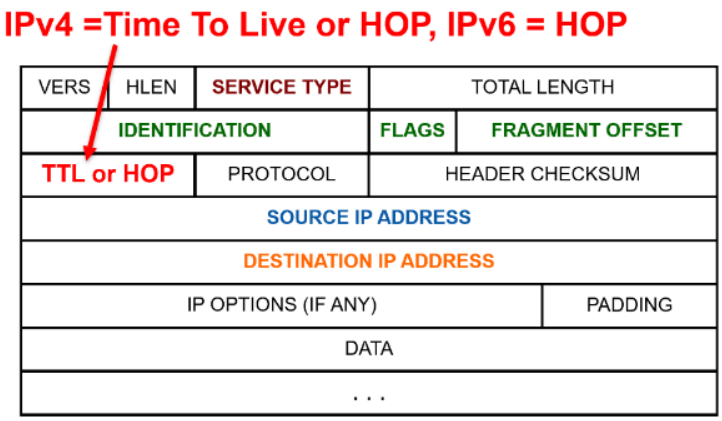
Hop limits were established to prevent “lost” packets from floating around the network forever. When a packet is transmitted, the hop limit is set to a default value. Each time the packet passes through a layer 3 device, the limit is decremented. When the limit reaches 0, the packet is destroyed by the layer 3 device holding it. This prevents a routing loop (a path that passes through the same devices without an exit) from forwarding a packet forever.

The default hop value for most systems is either 64, 128 or 255 but about 40 hops is sufficient to communicate anywhere on the earth. Even though there are millions of routers in the Internet, there are generally forty or fewer routers between any two locations.

Managing hop values controls the lifespan of a packet containing data. By limiting the lifespan of a packet from the protected endpoint, a security radius is created making it impossible for a packet to exit the limited perimeter. One example of hop growth was experienced by Microsoft in 1995. They had to change the default hop budget from 32 hops and chose 128 hops because network diameters were expanding requiring more hop budget to traverse more routers. Some devices were unable to exchange information because networks expanded and the default 32 hop budget prevented data from traveling through more routers to the destination. If network diameters continue to expand, Linux and others with a default hop count of 64 will require a greater hop budget value default to accommodate global data travel.

Figure 1: Packet lifetime know as Time to Live (TTL) or “Hop”

Figure 1 provides a breakdown of the IPv4 communications header used by all computers. The TTL or HOP field controls the number of routers that a packet can traverse and ensures the last router discards the packet when the HOP count reaches zero. The DATA portion or payload of the packet is not required by the HopSphere Radius Security method to determine and enforce safe communications lifespan limits.



I. Reducing Attack Surface in Simple Terms

The hop radius is essential to reducing the size of the attack surface, decreasing the number of attacks, and identifying attackers. For a given hop radius, then, the portion of devices in the world that can reach the server corresponds to the fraction of the world's devices within the corresponding hop distance. For example, a hop radius of 20 includes about half the world. 50% of the world's devices are within this hop radius and the other 50% are outside this radius. The exposure of the server grows nonlinearly with the hop radius as shown below.

A. Computers That *Can* Reach the Server

The number of computers that can reach a given device increase with distance. We measure distance in network hops rather than physical distance. There is an exponential increase in number of potential peers until we reach a point where there are no longer sufficient peers to maintain that growth. Growth then decays exponentially until all potential peers become within reach. A rough approximation of this behavior is given by the equation below (justification is shown in the Hop-Lobachevsky model).

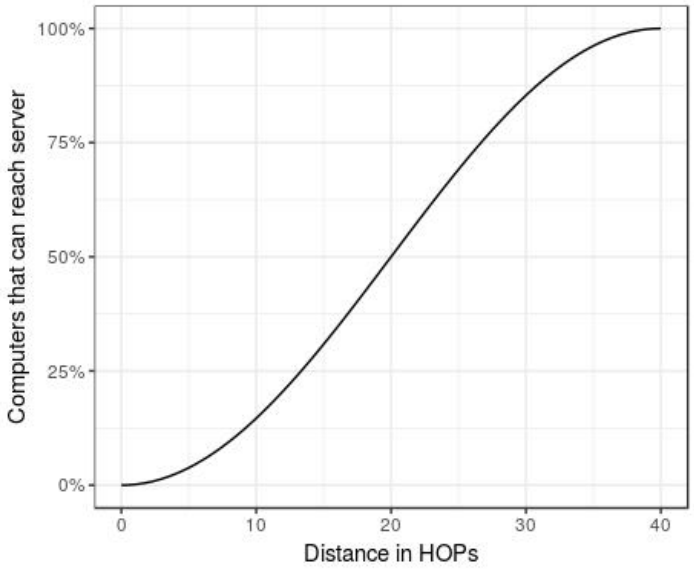
$$y = 100\% \cdot \left(\frac{1}{2} - \frac{1}{2} cos\left(\frac{\pi x}{r}\right)\right).$$

Where x is the number of hops and r is the maximum number of hops.

Or, using words,

Percent of computers that can reach server $= 100\% \cdot \left(\frac{1}{2} - \frac{1}{2} cos\left(\frac{\pi \cdot \text{Distance in HOPs}}{40}\right)\right)$.

Figure 2: Accessible Chart: Computers that *can* reach the Server



B. Accessible Chart: Computers Excluded and *Cannot* Reach the Server

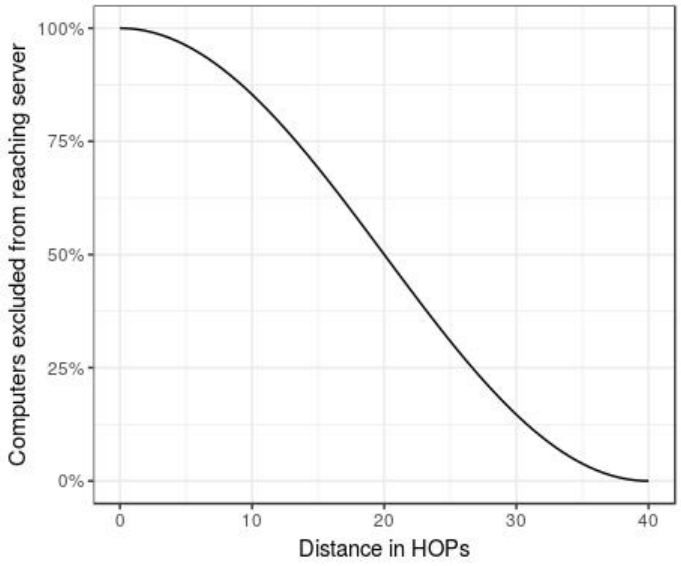
The percent of unreachable devices is the inverse of reachable devices (100% - the percentage of devices within reach). This equation can be simplified to

$$y = 100\% \cdot \left(\frac{1}{2} + \frac{1}{2} cos\left(\frac{\pi x}{r}\right)\right)$$

or

Percent of computers excluded from server $= 100\% \cdot \left(\frac{1}{2} + \frac{1}{2} cos\left(\frac{\pi \cdot \text{Distance in HOPs}}{40}\right)\right)$.

Figure 3: Accessible Chart: Computers that *cannot* reach the Server



C. Modeling Worldwide Accessibility

Using two models, Hops-Ptolemy and the Hop-Lobachevsky, we can illustrate the percentage of network devices that can be accessed or excluded from network data delivery.

Note: These models provide a visual representation of the impact hop limits have on data travel. When dealing with network topologies, the relationship between hops and geography can be very different. Devices separated by few network hops may be farther away geographically than other devices separated by many network hops. The emphasis here is on the percent of coverage. No attempt is made to imply which specific devices are reachable.

Model 1: HOP-Lobachevsky model

We use the Lobachevsky model to estimate the number of devices within a given hop radius. In this model, we assume our computer/server is placed at the North pole of the spherical Earth (the Earth has a form of a ball). The server could be anywhere on the spherical Earth, but for the sake of simplicity we rotate the Earth ball so the server comes to be at the top of the ball. We measure the distance from the server to another computer/device along the meridian. In this model the distance between the North pole and the South pole is 40 globe HOPs (measuring along a meridian). The distance from our server to the equator is 20 globe HOPs. The total Earth area is:

$$SS = 4\pi r^2$$

squared globe HOPs. The length of a meridian is 40 globe hops. The radius has length $\frac{40}{\pi}$. Substituting the length of the radius, we obtain the surface area in terms of hops:

$$SS = 4\pi\left(\frac{40}{\pi}\right)^2$$

For the sake of basic understanding and simplicity, our next assumption is that all devices are distributed uniformly on the Earth. It could look like the picture in Figure 4.

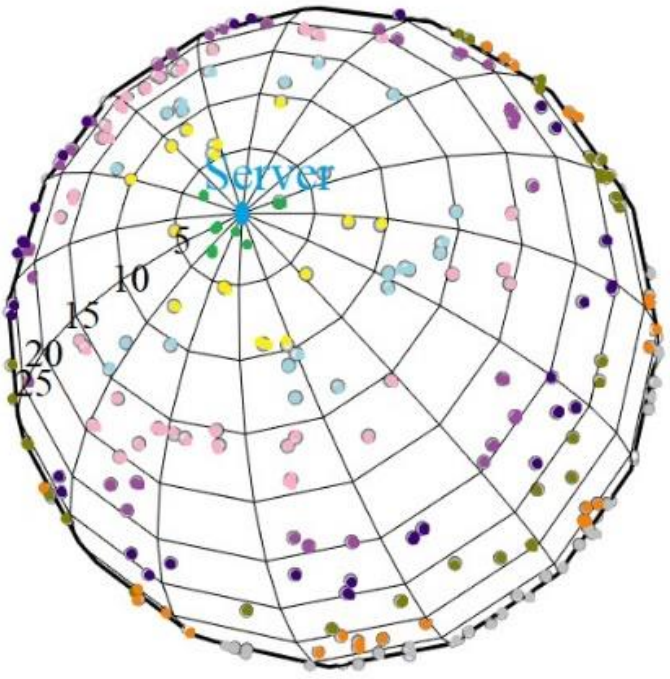


Figure 4: HOP-Lobachevsky Model Distance

Here we suppose that our server is at the North pole of the Earth globe (large blue dot). Other dots are the computers/devices in the world internet color-coded by distance from our server. The globe is covered by the grid of meridians and parallels. As an example, we pointed out the set of parallels as a concentric circumference on the globe surface - with the radii 3,6,9,12,15,18… of globe HOPs and so on.

HOP-Lobachevsky Model Distance – cont.

Let N denote the number of all computers/devices which are active in the world internet. We can say that in October 2014, Forbes Magazine reported that there were two billion personal computers in use throughout the world.[1] According to Statista.com, in 2018 there were 23.14 billion connected devices (computers/mobile phones, etc.).[2]

The uniform distribution of the dots (computers/devices) on this globe means that each one squared globe HOP (not a flat squared HOP) contains:

$$\frac{\text{total amount of computers,devices}}{total\ globe\ area} = \frac{N}{SS}$$

i.e. about D = $\frac{N}{SS}$ computers/devices are in a one squared globe HOP. It gives us a clear idea how to get the number of computers, devices in a concentric circle/disk (on the globe) with the radius r globe HOPs:

- in a one squared globe HOP there are about $\frac{N}{SS}$ computers, devices
- then in the circle/disk (on the globe) with the radius r:

  $$\text{area} = 2\pi\left(\frac{20}{\pi}\right)^2\left(1 - cos\left(\frac{\pi r}{40}\right)\right)$$

  there is a proportional quantity of the computers, devices, i.e.

  $$\frac{N}{SS} * \text{area} = D * \text{area}$$

If we protect our server from computers which are farther than r globe HOPs away, the number of computers which can reach our server is the number of computers within the circle/disk/spherical cap/spherical segment (on the globe) with the radius r globe HOPs, i.e., reachable computers.

Figure 2 is a function graph (which is a curve in trigonometric terms) showing how many computers can reach our server depending on the protected distance (in globe HOPs). For simplicity, we operate with the percent of the total number of computers, devices and not with the hypothetical D. We can get the formula as:

$$\left(\frac{1}{2} - \frac{1}{2}\cos\left(\frac{\pi r}{40}\right)\right) * 50\%$$

Consequently, all computers which are farther than the protected HOP distance are unreachable from our server and thus cannot reach our server. The trigonometric curve (shown in Figure 3) is complementary to 100% of the reachable server graph. The equation for this chart is:

$$\left(\frac{1}{2} + \frac{1}{2}\cos\left(\frac{\pi r}{40}\right)\right) * 50\%$$

Model 2: HOP-Ptolemy model

The Ptolemy (flat earth) model provides another way to visualize how hop limits impact device reachability. In this model our protected device is a center of the flat Earth which has a form of a circle. The radius of the circle is R, the maximum distance between devices, which we have observed is about 40 hops from customer locations. The area of this circle is a total area of the Earth.

For the sake of simplicity, our next assumption is that all devices are distributed uniformly on the Earth circle. The number of devices within a unit square is given by taking the total number of devices N divided by total surface area S. Since devices are evenly distributed, the number of devices within any given hop radius is determined by the area within the radius:

$$\text{area} = \pi r^2$$

We already showed the percent device reachability at a given hop radius can be found by:

$$\left(\frac{1}{2} - \frac{1}{2}\cos\left(\frac{\pi \text{ hops}}{40}\right)\right) * 50\%$$

Note the percent of devices is not linearly related to the number of HOPs. We can compute the % unit radius r for each hop limit by solving the following equation:

$$r(hop) = \sqrt{\%\ area(hop)}$$

In Figure 5, the blue dot in the center represents our server. The red dot is surrounded by concentric circles corresponding to various HOP limits. Dots are color-coded to represent HOP distance.

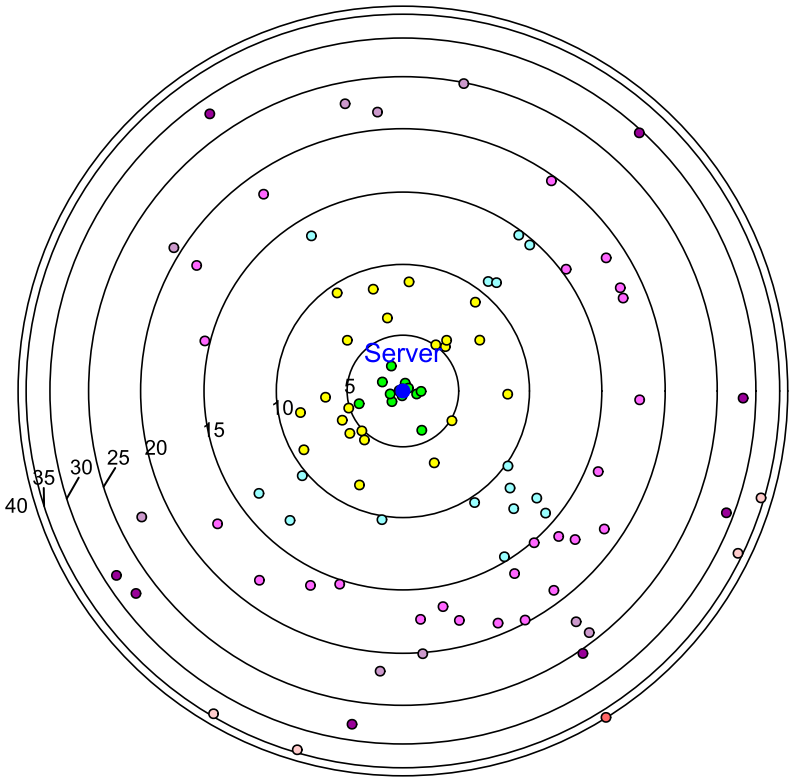


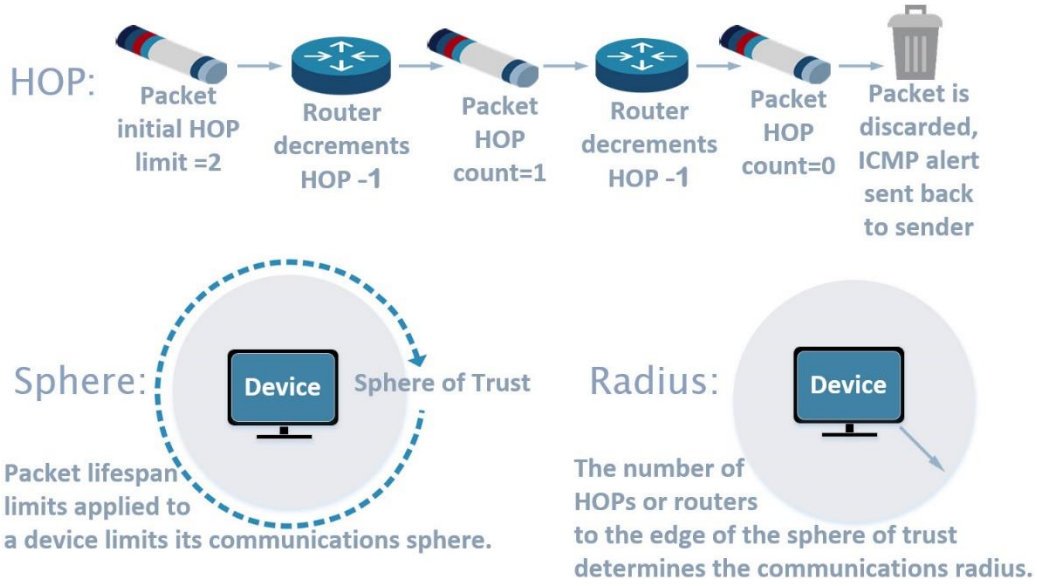
Figure 5: HOP-Ptolemy Model Distance

Note the nonlinear spacing between concentric circles in this model.

II. Overview of HopSphere Radius Security

The diagram below illustrates the basic concepts for HopSphere Radius Security.

Figure 6: HopSphere Radius Security Overview



In Figure 6, as the packet starts its journey from a computer, it has a default travel budget. In this example we changed budget to a value of two (2). It traverses the first router and the hop budget is decremented from two to one (1). At a value one (1) it cannot successfully go through another router without the next router decrementing the value to zero (0). When any router encounters a hop budget of zero (0) the packet is discarded. After discarding the packet, the router sends an Internet Control Message Protocol (ICMP) packet alerting the sending station of the discard. Inside the ICMP is important information containing a copy of the original packet that was discarded for reference. HopSphere Radius Security decodes the message learning the source and destination and other valuable information to help report the event which is eventually sent to the Security Operation Center (SOC) for action.

HopSphere Radius Security creates a secure perimeter allowing appropriate communications to occur between devices inside a sphere of trust. When an attempt to communicate outside a sphere occurs, the packet lifetime (hop) reaches zero the router destroys packets destined beyond the limited perimeter. In the diagram above Server A's Hop value is changed in the devices configuration to use a lower lifetime value that prevents packets from transiting or "hopping" through the router when the hop reaches zero. The router seeing the hop decremented to zero discards the packet. Secondarily, the discarding router sends an alarm message back to Server A informing of the discard.

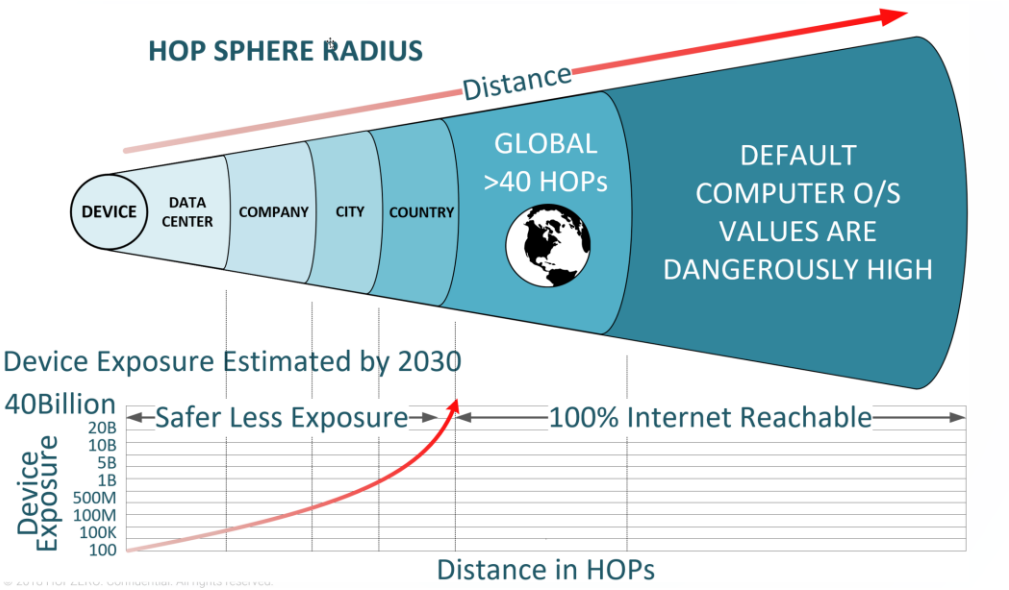
The result of this standard, unmodified behavior of all routers, cyber-attackers with stolen security credentials are blocked from making connection and thereby unable to get a login prompt from outside the protected radius. Login prompts are not provided by the protected

endpoint because one direction has a limited packet lifetime value. An attempt by cyber-attackers outside the HopSphere Radius Security System creates an alert send to the Security Event Information Management (SEIM) management console operated by the Security Operations Center (SOC). The alarm contains the IP address of the attempting station identifying the attacker and is stored as evidence of the attempt for remediation and potential legal action. The attack was successfully blocked and no compromise occurred.

III. Conclusion

Not all devices or servers should have a lower hop value. Web servers that want to have global visitors for instance want to keep the default communication values. Sensitive databases that need only communicate inside the datacenter require the correct value to be learned, set, enforced and be monitored for access attempts. Leaving the global default hop value on sensitive servers allowing them to reach or be reached globally significantly increases risk. HopSphere Radius Security reduces the threat of data compromise by changing the current security paradigm. Lowering packet hop budget from global travel defaults greatly reduces the attack surface exposure for those systems able to operate in a smaller sphere of trust. Doing so protects valuable data from exfiltration, capture and ransom. Figure 7 displays two important views of network data travel. The cone diagram shows how lower hop values can keep information inside the datacenter or organization, safe from the Internet's reach. One only needs to use a few hops to limit travel to the datacenter or organization. We do not advocate using a hop limit within the Internet as hops vary significantly. The Device Exposure chart y-axis shows the projected number of devices on the Internet by 2030 at 40 billion devices. The x-axis shows how lower hop values reduce the number of devices that can reach or be reached by a device. HopSphere Radius Security allows one to choose the level of exposure risk as demonstrated by the Device Exposure chart below.

Figure 7: Default Computer O/S Values Allow Global Data Travel



About the Authors

William “Bill” Alderson has practiced catastrophic IT critical problem resolution services since the 1980’s, providing training and services for 75 of the Fortune 100 companies and has certified 3000 network security forensic professionals. Bill gained notoriety when his team was called to the Pentagon immediately following 9/11 to help diagnose catastrophic IT problems. He also performed many early zero-day security diagnoses - including US Stock market DDOS. Bill has had two corporate acquisitions and one IPO, was at Network General Sniffer in 1986, and PMG was acquired in 2005 by NetQoS/CA Technologies serving as a technology officer. Bill is the CTO and CEO of HOPZERO, Inc.

Ben Haley, has earned an MBA and Masters in Computer Science and was initial development manager of NetQoS, network performance analytics company. Oversaw architecture and led research team in detecting network anomalies. NetQoS was acquired by CA Technologies in 2009. Ben led software development for new markets at MaxPoint Interactive, a large online advertising company. This involved complex, high-speed, high-throughput analytics. MaxPoint was acquired by Valassis Digital in 2017. Ben is the SVP Development of HOPZERO, Inc.

Reference & Attribution

1. “How many things are currently connected to the internet of things?”, Forbes Magazine, 01/07/2013. https://www.forbes.com/sites/quora/2013/01/07/how-many-things-are-currently-connected-to-the-internet-of-things-iot/#6a5f141abd2d
2. “IoT number of connected devices worldwide”, Statista.com. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/
3. Judy Ann Michael, MBA, editorial services, JudyAnnMichael.com.

HOPZERO, Inc

11400 Concordia University Drive
Austin, Texas 78726
Inquiry@hopzero.com
www.hopzero.com
833-467-9376