How the Internet Works, Part 1

Description: During this 49-minute episode, Leo and Steve briefly discuss the "Kama Sutra" virus that will become destructive on February 3rd. We briefly discuss PC World Magazine's recent evaluation and ranking of ten top anti-malware systems. And we begin our long-planned "fundamental technology" series with a two-part close look at the history and detailed operation of the global Internet.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-025.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-025-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 25 for February 2, 2006: How the Internet Works, Part 1. Steve Gibson, and a very good day to you, sir. How are you today?

Steve Gibson: Hi, Leo. Great to be back with you.

Leo: I want to thank you for sending me that great sci-fi book. Tell me the name?

Steve: It's called "Fallen Dragon."

Leo: "Fallen Dragon."

Steve: Yeah. Peter F. Hamilton is my current, my favorite sci-fi author. I've read everything he has in print. And in fact I'm rereading the last book I read of his because it's been a couple years, and he's coming out with the second part of that book, which was - the first one was called "Pandora's Star." Anyway, I just, you know, I love sci-fi. And you can't find enough movies and TV, so I read when I'm not doing anything else.

Leo: Well, as you know, I'm off on a cruise, and I just needed a book. So the timing couldn't have been better. I can't wait to read it.

Steve: Yeah, "Fallen Dragon" is a really nice novel.

Leo: I'll give you a review when I come back.

Steve: Can't wait.

Leo: Meanwhile, we're going to talk about how the Internet works in just a little bit. But I've got to ask you about Kama Sutra...

Steve: Yup.

Leo: ...which is not the Indian book of love, but in fact a virus that is expected to hit hard on Friday the 3rd.

Steve: Well, yeah, the good news is it's been known for a couple of weeks. It was first - it first appeared on...

Leo: How do we know about it?

Steve: ...January 20th. Well, you know, people began getting it. And interestingly enough, every instance of the virus that infects a machine goes and pokes a website that has a web counter on it. So by looking at the counter, people have been able to determine approximately how many copies there are.

Leo: Some hacker just wanted some glory, didn't he.

Steve: I guess. You know, who knows? What's significant about this virus, and the reason we're talking about it today - and we don't normally talk about viruses, I mean, you know, there's a hundred new ones a week, literally - is that this is a destructive virus. Most current viruses and worms have the goal of getting into someone's machine for the purpose of taking it over for spamming or using it as a denial-of-service attack launch platform or something. This is the first virus or worm we've had for a while where it's deliberately destructive to the user's computer. What it does is it overwrites the contents of DOC files, Excel spreadsheets, PowerPoint presentations, ZIPs, RARs, PDF files, and MDB databases. So, and it doesn't just delete them, it actually fills the file with garbage so it's less clear what's happened. And, of course, if you didn't notice it happened, and you were doing regularly scheduled backups, you might overwrite your good backups with now garbage-filled, you know, newer versions.

Leo: Ugh. This is like the old days of viruses, when they really used to all be destructive. They haven't been in a while.

Steve: That's true. Because, again, the whole new goal is to acquire these botnets and fleets of things which the hackers, the malicious hackers then rent out for, you know, in order to get payment. So on our show notes site, on our show notes page for Episode 25, I've got a link to - I sort of summarize what this is all about. It launches 30 minutes after you start your computer on the 3rd of any month. And so tomorrow, being February 3rd, will be the first launch event for this - it's called Kama Sutra; it's called Blackmal. It's actually a flavor of a worm/virus that's been around for a couple years. But this latest incarnation of it has changed its behavior in this very malicious fashion. So it's also called Nyxem. I don't know how you pronounce that.

But Symantec has a very nice free little scanner that I have a link to on our show notes page. It's easy to download. You don't have to register yourself or give them your email address or anything. It's very clean. You can grab the scanner, run it on your machine. Now, again, I don't want to overhype this at all because all AV companies have known about this for a couple weeks, since it was first discovered several weeks ago. So it's probable that, you know, no one who's security conscious and is updating their AV patterns consistently is going to have a problem with this. But if our listeners know of people who have had problems with viruses or aren't that careful, it might be worth them telling their less security-conscious friends to run this little scanner just to make sure because it is, if it hits you, it's potentially damaging.

Leo: By the way, somebody asked me - and I do the same thing, I recommend these removal tools or these standalone scanners from Symantec - they said to me, well, you hate Norton Antivirus, so how is you recommend Symantec's removal tools?

Steve: It's just clean. I ran across it, I tried it today, I recommended it to both of my employees, and I sent out a very small little email to people who I know, you know, it wouldn't hurt them just to run this. So...

Leo: And as I point out, there's a big difference between a small, one-time-only program that does this

and a big, bloated, huge program that takes over your system, so...

Steve: Right. I mean, you know, unfortunately Symantec, I mean, I don't want to criticize it to people who are sensitive to and love Symantec. But, you know, I like small, lightweight solutions.

Leo: Right.

Steve: And Symantec is widely regarded in the industry as having really gone the other direction. It's just huge.

Leo: But their labs are still good, their virus researchers are still very good, and the removal tools they write are absolutely great. They do the job.

Steve: Right. Well, speaking of viruses, while we're on the topic, PC World just released their March 2006 results for malware detection software. One of the questions I get constantly, and I know you do, too, Leo, is what's the best AV? Well, just answering that question or, like, doing the research myself is the last thing I'm ever going to do ever.

Leo: Oh, it's impossible.

Steve: Well, it is.

Leo: There's too many viruses, too many programs. You just need a lab to do it.

Steve: Well, PC World has done a very nice job. And also on our show notes page I have a link to their comprehensive review. What I will tell people, and I'll just sort of for the fun of it read this off from best to least good, their best-buy choice was BitDefender 9 Standard Edition, which got their top rating, a score of 92. And they called it "superior." It's the number one choice for, like, what they were looking for. And they have charts and, you know, all the details. Number two was McAfee's VirusScan 2006. Coming in third place was Kaspersky Labs' Anti-Virus Personal 5.0. Then F-Secure Anti-Virus. Then Symantec Norton AntiVirus 2006 in fifth place. Panda Software was in sixth. AntiVir Personal Edition, seven. Something called ALWIL Software avast! Home Edition 4.6 was in eighth place.

Leo: That's a free one, avast!. A lot of people use that, yeah.

Steve: Yes, yes. And PC World said it was good and gave it a 77 rating. Trend Micro's PC-Cillin took ninth place.

Leo: Really, below either of the free ones, AntiVir and avast!. That's interesting.

Steve: Yup.

Leo: Wow.

Steve: And it's $50, whereas BitDefender is $30.

Leo: Wow.

Steve: The number one ranked by PC World. And then finally AVG's Free Edition is in last place, number ten.

Leo: I'm a little disappointed they didn't review my favorite, NOD32, but I guess they left it out.

Steve: Good point.

Leo: There are other antiviruses, although their top five are all well known. I'm not a big McAfee fan. I thought that was interesting that came in second, but...

Steve: Yeah.

Leo: It does a good job; it's just another one of those big programs, you know?

Steve: Right, right. It and Symantec.

Leo: So links on your show notes at GRC.com/securitynow.htm to this article, so people can read the details.

Steve: Right.

Leo: Now, is it time?

Steve: Yup, finally.

Leo: How the Internet Works, Chapter One.

Steve: This is the beginning of a series we're going to do deliberately to lay a bunch of foundation. I mean, I'm going to try to really explain everything so that, when we talk about specific issues relating to security - and of course I'll explain it while we go along in the context of security, since that's the whole point, not just to talk about bits for bits' own sake. But I want to lay down a foundation for, you know, what these terms use that we've already, for the last 24 episodes, been using glibly. And most people understand them. But what I find is that, when people write in, even, like, you know, network directors for major IT companies, they say, yeah, I pretty much know generally what you guys are always talking about. But I always pick up something that had never occurred to me, or a little corner that I hadn't gone into before. So...

Leo: So a review of the basics is good for everybody.

Steve: Right. So the Internet. When I first heard - and this is in the early '80s, I think, is when it first really became apparent that people were talking about the Internet. You remember back then we all had modems, and we were dialing into The Source, or CompuServe...

Leo: CompuServe, GEnie, yeah, yeah.

Steve: Exactly. This notion that we were going to network all of the machines in the world or in some large space together just seemed loony to me because, coming from a hardware background, I thought, okay, what about the wires?

Leo: Yeah, loony, it seemed impossible.

Steve: You know, how are we going to wire this all together?

Leo: And not to mention, I mean, at that time, if you had MCI Mail, you couldn't send mail to somebody on GEnie, and they couldn't send mail to somebody on CompuServe. They barely had gateways to one another, let alone talking to one another.

Steve: Right. And in fact the breakthrough in conceptual approach was - and this happened in the mid-'60s, actually, first at MIT, and also at Stanford Research in Palo Alto - was this concept of switching from the idea of a circuit to a packet. And that was the key. You know, I don't think there's been a single Security Now! episode where we haven't talked about packets - packets this, packets that, IPs and ports and, you know, and packets. And a packet is the fundamental, sort of like the DNA of the Internet. And it was that concept that people had that instead of actually, for example, running wires between you and everyone you want to talk to, instead you could have an architecture where your data would go through intermediate spots that were richly interconnected, and one way or another these packets of data would get to their destination. That was the key. And in fact, you know, and that was the shift in thinking that really made the Internet possible.

You know, when I first heard, oh, we're going to connect all the computers together, it's like wait a minute, you know, if you have N computers, you've got N times N minus 1 interconnections, that is, if you connect every computer to every other computer. And so of course the beauty of the Internet is you don't need to do that at all. You need to connect your computer to some other computer, or maybe to a so-called router, which is connected to another one, connected to another one, and connected to another one. And with enough of those intermediate cross connections, your data can reach any other machine on the planet. And that's how the Internet works, basically.

Leo: It's really a miracle. I mean, the phone system for years was a switched network, where you had essentially a wire stretching between you and the other person on the other end.

Steve: And a whole bunch of wacky copper contact relays.

Leo: Or an operator who said, "Number, please."

Steve: Right.

Leo: So all of that was just kind of eliminated by this notion of packets. A brilliant idea.

Steve: So overall, what we're using today, and the only thing that most people feel, is called Version 4, IPv4, or Internet Protocol Version 4. 1, 2, and 3 were sort of starter protocols that didn't have much life. When we got to 4, it was a rich enough solution that, you know, we've had it ever since. Essentially, all the Internet that anyone knows has always been Version 4.

Now, for strange historical reasons, Version 5 isn't ever going to happen. We're essentially going to eventually move to IPv6, Version 6. IPv6 isn't something that at this moment we're going to ever spend much time talking about because it just hasn't happened. One of the concerns that was driving Version 6 was people were worried we were going to run out of addresses on the Internet. It's funny, when the very first designers of this started up, they said, I mean, they had no concept that people would be walking around with computers in their pocket that were part of some global network, which is of course what we have today. They were envisioning maybe national-size networks of, like, universities or major corporations. When they laid out a 32-bit address, their idea was that the top byte, the first byte of the address, would be the network number, that is, essentially the location number. And the other 24 bits of the 32-bit address would be the machine at that location. So like a university would be 3, and then followed by which machine at that university. And a different university would be 4.something.

So that was the original concept was that they had the idea of huge local networks that were then able to

reach across to another different local network using this Internet protocol where the first number was which other network you wanted to go to. There was, you know, nothing like what we have today was envisioned. And so it's really something of a miracle that something that was designed so opportunistically 20 years ago, even a little more than that, actually, has survived as well as it has. So what IPv6 will do is it expands that 32-bit IP address that we'll talk about here in a second in great detail, it expands that 32-bit IP address to 128 bits. Well, 32 bits is 4,294,967,296.

Leo: You can see why they thought that was a lot.

Steve: Well, exactly, even though they weren't even going to use it that way. They were just going to use the first byte, which can be 0 through 255, as, like, which big meganode somewhere in the country the packets should all go to. Then they'd go to a particular machine at that university or corporation. So we have, you know, 4.3 almost billion IPs currently. Well, 28 bits for addressing, which is what IPv6 gives us, is really out of control. That's 3.4 times 10 to the 38th power. That's 340 billion billion billion billion IPs. So...

Leo: That should be enough, at least until we conquer a few more galaxies, I think.

Steve: Yeah. We don't have to worry about that. The problem is that it's not easily compatible with IPv4. The problem that IPv6 is having is, you know, the manufacturers who are making the routers, I mean even, for example, the PC manufacturers are supporting Version 6, though no one's using it yet. You know, Windows Server 2003 and XP can do IPv6. But you can't get it anywhere. I mean, there's nowhere to plug it in to get Version 6. Now, I mean, that's not strictly true. There are university environments which are sort of using it experimentally. And in fact what's happened is that, in order to straddle the address change, IPv4, our little 32 bits, sort of tucks itself in one little corner of IPv6 so that you're able to transport IPv4 data across an IPv6 network. It doesn't get confused and lost. You are able to do it. But for all intents and purposes, it just hasn't happened yet.

Leo: So is it just going to take so much to implement that it will never happen, or are people moving in that direction? I mean, whither IPv6?

Steve: What's happened has been really interesting, and that is that the NAT router technology that we talk about so much because it's such great security for typical home users, you know, home users are sharing many IPs in their home behind a single IP at their ISP, that is, a single public IP. So that's a perfect example of what's happened. If it weren't for NAT router technology that basically allows many machines to share a single public IP, we really would be in trouble already with IP space depletion. But NAT routers happened, and they're just a good thing for everybody. Corporations are using them. There are even some ISPs that are using NAT routers and putting all their customers behind a big NAT router because it really works very well, not perfectly, but very well, as most home users know. And so the prevalence and birth of NAT routing technology has hugely reduced the pressure on the move to IPv6.

Now, IPv6 is more than just bigger IP numbers, more than just this move from 32-bit IPs to 128-bit. It does offer also some inherent, what's called "link-level security," or IP-level security, where you're able to encrypt your endpoint-to-endpoint connection using the protocol, rather than having to lay it on separately as we've been talking about before, you know, during all of our talk about VPN. So there are some other features which are beneficial, but they're just not driving forces. And now that there's no big rush to IPv6, due to the fact that we're not about to run out of IPs, it's really not clear when we're going to make the move. It's the kind of thing, you know, it's expensive to do. People have to remove equipment. And, you know, for example, all of the NAT routers that we have installed right now are IPv4. They're not IPv6. Probably the same is true with the cable modems, although I'm less sure of that. So my point is that we're talking about a massive obsolescence when we finally do move. It's sort of like, you know, how the FCC has been trying to push us to HD-TV, and ultimately they're going to be turning off the NTSC over the air to force those last people who've got, you know, rabbit ears sticking out of their TV to finally move.

Leo: If you don't do it, you know, if you don't turn it off, nobody's going to change, frankly.

Steve: Right. But no matter what, the essence of the Internet are these things called "packets." Now, a packet is nothing but a block of data. So the first thing, I mean, just in terms of demystification, a packet is nothing but just a blob of binary bits. The Internet protocol defines the layout of these packets. And at sort

of like the outer layer you've got a source IP and a destination IP, which as we've talked about are 32 bits uniquely specifying where this packet came from on the internet and where it's going to.

Now, in the original sort of pre-NAT version of the Internet, packets actually did contain the IP address of the machine. We know that's no longer true because, for example, if - well, for example, right now, Leo, you and I are conversing over VoIP. Both of us are behind NAT routers. It happens that my computer is the IP address 10.0.0.10. Yours is probably 192.168.0.4 or 1 or something.

Leo: Right.

Steve: So those are both private IPs that are nonroutable. So the packets which are flowing between us across the Internet do not identify our machines, they identify our NAT routers.

Leo: Although that is a private - or not a private, a public address assigned to us by our Internet service provider. I mean, it's a real address.

Steve: Correct.

Leo: Yeah.

Steve: But I guess the point is that...

Leo: But there's one per person instead of one per computer.

Steve: Exactly.

Leo: Right.

Steve: And it's when the packet arrives at our router that the router then does sort of the next level of job of figuring out which machine behind it has an established connection with the other endpoint. And so it then routes it inside of our local area network to the proper machine. So it's not the case any longer that IP addresses uniquely identify machines. They identify sort of like the public endpoint. And at that point the network may go private, as it does when it crosses into a NAT router. Even many universities are doing this.

So, you know, people have asked, when they're concerned about file sharing and, you know, NSA spying, and how it is that their IP addresses can be tracked down to them, it's certainly the case that an IP address owned by an end user can be identified, you know, like by their household. But once it goes into your NAT router, unless the router is logging specific IP assignments behind it - and most of them typically don't; but, for example, universities and ISPs do. Unless there was that, there's really no way for anyone to trace it beyond when the IP stops being public and switches into being a private IP. There's no way to know where it goes.

Leo: People are using something called DHCP to assign those addresses. I do remember in the earliest days of these routers people would sometimes use static IP addressing and buy five Internet addresses and just use the router to kind of bridge, almost as a hub, really, I guess.

Steve: Well, yes. And in fact, the earlier days of DSL you would be able to get from your phone company, you could get a DSL network with eight IPs.

Leo: Right.

Steve: And you'd actually have eight physical, public IPs as opposed to just one. But again, because they're so scarce, it's really increasingly hard to do that. And in fact, even people like myself, where I've got a corporate network, you have to now fill out - and I had to do this less than a year ago - what they call an "IP Justification Sheet," which they have on file in case they need to explain why they need the IP space they have.

Leo: Right.

Steve: I mean, it's really becoming a scarce resource. And we're not running out. But, you know, there's already people being more responsible about the way these IPs are being handed out.

Leo: Do you think we'll ever see IPv6 implemented?

Steve: Well, I mean, it's scattered around. You can connect equipment together, and it does work. But it just, you know, in order for it to work from endpoint to endpoint, every single piece of equipment has to be IPv6 compatible along the way. And we're, you know, we're a long way from being there. It's just - I really would be surprised if it happens.

Leo: We've figured out a way around it.

Steve: Right, right. Okay. So we have packets, which have a source IP and a destination IP. The other thing that a packet, a standard Internet protocol packet also has is what's called a "protocol number." Now, that's an eight-bit byte that tells the system or routers, basically anyone handling this packet, what type of packet it is. And, you know, we've talked about ICMP, you know, pings, and TCP and UDP. Well, all those things are are just different bytes. ICMP happens to be a Type 1. And TCP is a Type 6. And UDP is a Type 17. So the only thing that really differentiates sort of these otherwise generic Internet protocol packets is this one byte up at the front of the packet that says the rest of me is this type. And so that's how equipment that is having to deal with this knows what to do with it. ICMP is sort of used for plumbing maintenance. If links go down, that is, between two points, you're no longer able to ping the other endpoint, it's used by sort of the mechanics of the Internet, sort of not as much for moving actual data, but for sort of managing the transport of the data and being able to check which equipment's online and for various sort of supervisory functions.

One of the other interesting things that's in every Internet packet is called the Time To Live, or the TTL. What you have with a whole bunch of routers all linked to each other - and we'll talk about how that works in a second - what you have is basically one router passing the packet to another router that then passes it to another and so on until it finally reaches its destination. Well, there is the possibility for so-called router loops where, for example, if one router were misconfigured, it might send the packet, instead of sort of sending it further towards its destination, it might send it upstream by mistake to a different router, which would then receive it and, just like it does any other packet, send it back towards its destination. Well, due to misconfiguration, you could end up with loops between routers so that one poor packet just sits there being handed off in a circle, never being able to reach its destination due to a misconfiguration.

So the original designers recognized the danger of this, and so they put a Time To Live in each packet. Even though it says "time," it's really not about time, it's about router handoffs. It's a counter. And so systems will start off setting the Time To Live, for example, to 64 or 128 or some semi-arbitrary value, whatever they're configured, to sort of launch packets out of themselves at. And every time a router receives a new packet, it decrements that Time To Live counter by one. As long as it doesn't go to zero, it considers that the packet has not expired. And so it looks at the destination IP, figures out which of its many multiple connections is, like, going to take that packet closer to its destination, and so it sends it on out of itself to the next router in the chain. That router receives the packet, again decrements that TTL by one. And again, as long as it still has some life left in it, literally, as long as that counter hasn't hit zero, it forwards it again.

So the beauty of this approach is that, if a packet ever got stuck in a router loop, those routers passing it back and forth by mistake among themselves would each be decrementing this TTL towards zero. Finally one of the routers - sort of like playing Hot Potato - one of the routers would decrement it from one to zero, and at that point the packet is said to have expired. What the router then does is it - because this is like abnormal behavior, you don't want your packets obviously to be expiring - it will take the packet and sort of the first chunk of the packet and wrap that in one of our - what we were talking about before, an ICMP message called Expired. It wraps it in an ICMP Time Expired message and sends it back to the IP that apparently put that packet onto the Internet, just to inform the sender that, hey, we don't know why, but we

couldn't get your packet all the way to its destination before it expired.

Well, one of the really interesting things that was happening actually only a few years ago is that systems were not configured with the present and growing size of the Internet in mind. There were some computers, in fact the early Windows machines, that would set their packets' TTLs, their Time To Live, to 32. Well, back then, 20 years ago, 32 was enough because what that would mean is that, if you were sending it from a source location to a destination, you could go through 32 router handoffs before reaching the destination. Well, the Internet not being that big, sort of in terms of like the diameter of the routing, or the so-called routing diameter of the Internet, you would be able to get it there. But as ISPs became more internally complex, as more ISPs were born, and they hooked their routers to upstream routers, what happened was the so-called routing diameter of the Internet grew to the point that it became larger than 32 routers. And what that meant was that there were some people who were using earlier versions of Windows or other operating systems that were still using a value, a starting value of this Time To Live that was too low. They were unable to reach other locations. Their packets would die, even with no routing problems at all. Just, you know, 32 hops, and they hadn't quite gotten to their destination, and that final router would say, uh, sorry, and send it back to the person who originally put that onto the Internet, saying this thing died in transit.

So one of the things that we've seen has happened recently is that, as Windows and other operating systems have been run through their versions, this TTL has been bumped up to 64 and then typically to 128, sometimes up to its maximum value of 255, which, you know, if we ever get more than 255 routers between two points, we're really in trouble because no data will ever be able to get from one end to the other. Isn't that cool?

Leo: I really like the elegance of the solution. How much of it is planned ahead? How much of it is in response to problems as they come up?

Steve: That's one of the things that is shocking is that so much of this was done right the first time. I mean, even though these guys had no idea where this was headed, I mean, you know, 25 years ago you couldn't have imagined this. You know, even so, they laid down these fundamental protocols and got it right the first time.

Now, one really cool application of this packet expiring is something that many people who've messed with networking for a while have used and may not have really appreciated what was going on, and that's called the traceroute. Because when a packet expires on its way to its destination - remember I said that the router where it expired, the router that decremented that packet's Time To Live from one down to zero, it wraps the packet in a little ICMP message and sends it back to the sender. Well, that means that the packet is being sent from that router, from that router's IP, back to the sender. So the neat engineers who thought up all this stuff in the beginning, they said, hey, what if we wanted to know the route that our packets are taking as they go from one router to the next across the Internet? How could we determine that? Well, what they realized was, hey, we could deliberately emit packets with shorter Time To Lives, so that on purpose they die before they get to their destination.

So what the so-called "traceroute" command does is it sends the first packet towards its destination, like say you were tracerouting to Microsoft.com. It would address the packet with the IP address of Microsoft.com, but set the Time To Live on purpose to one. So the first router that receives it decrements it to zero and goes, whoa, this must be an old packet. Well, so it sends a Time Exceeded message back to the sender with its IP. So we display that, and we now know the IP of the first router on our path towards Microsoft. Then we send out a packet with a Time To Live of two. Well, it goes past the first router, which decrements its time to one, gets to the second router, which decrements its Time To Live to zero, and now that sends it back to us. So by repeating that process until we finally reach Microsoft.com, we end up with a very elegant little listing of every router between us and our destination. And of course that's useful for all kinds of problems. And in fact it can be used for many Internet diagnostics purposes.

Okay, now, the last thing I want to cover in this Part 1 is about routers. You know, we've talked about how packets move from router to router. The question is, how do routers know where to send it to? And this is, I think, another one of the core bits of genius of the original designers of this thing, of the whole technology. If IPs were assigned at random so that any IP could belong to anyone, routers would have a horrible problem. They would literally have to have a four billion long table...

Leo: Oh, okay.

Steve: ...that had everybody's IP and which interface, which of their interfaces is connected one way or

another eventually to that user.

Leo: Well, the Internet wouldn't work if you had to do that.

Steve: No, it just wouldn't work. So instead, the IP space, this 32-bit IP is inherently a hierarchical addressing system. Now, we're very familiar with hierarchical addressing systems because, for example, that's how postal mail is routed around the United States. You know, my mailman doesn't know where Leo lives. But he knows where, you know, the town you live in is.

Leo: Right, right.

Steve: Maybe, you know, just by using ZIP code, for example.

Leo: Well, and they're hierarchical, too, aren't they.

Steve: Well, exactly.

Leo: I mean, 9 is California, or is the West Coast, I guess, and...

Steve: Well, I guess it's sort of linear. But you're right, it certainly gives you a perfect starting point. And so, for example, if we think of addresses as in, like, which state in the United States, then which town in the state, and then which street in the town, and then which number on the street, you know, there's a hierarchy of state, town, street, number, which almost exactly mirrors the way Internet protocol works. So, for example, you know, there might be a Laguna Hills, Florida, and there certainly is a Laguna Hills, California. Because they are in different states, they're not confused.

So the way the Internet protocol works is very similar to this. When an ISP is given an allocation of IP addresses, it gets a certain number of sort of like high octets, like, for example, you know, cable modems for a long time tended to be 24 dot. And I think now a lot of them are in the 68 dot and 67 dot, meaning that it's, you know, 24.x.y.z, whatever the other numbers are. So the idea was that a large cable modem ISP would get all of the addresses that start with 24 dot. So what that means is that some router, you know, at the other side of the world, all it has to know is that any IP address beginning with 24. is sort of in that direction. It just sort of - all it has to know is kind of over there, you know?

Leo: Right.

Steve: You know, if you imagine that it had four interfaces - north, south, east, west - going to other routers to the north of it, to the south of it, to the east and the west of it, then its job would be to get an IP packet, look at the destination IP, and look up in a much shorter table. Now its so-called routing table, if it had an entry that began with 24, it would just say, oh, 24, you know, dot star, essentially, you know, to use the terminology of, like, of a wildcard, anything beginning with 24 I send west. And anything beginning with 4 I send north. And anything beginning with, you know, something else, I send in a different direction.

So it's actually the excruciating detail level is a little more complex than that. But that's the whole concept of routing that makes this whole global network function, is routers talk to each other - which is another very complex topic I think we'll probably never get to - using something called BGP, Border Gateway Protocol, and a couple other protocols. They're able to share with their neighbors the IPs that they know about and the connectedness of their other interfaces, so that their neighbors know what packets to send to them. But basically this forms a big grid of routers, each that only have to know, when they receive a packet, which interface that they're connected to that they should send that out of in order to move it in the direction it's going to. So in our example, where we had a packet that was addressed to 24.something, it would march across the globe from router to router that is just looking at the packet, going, whew, all I have to do is worry about this 24, and that goes out my, you know, interface number three. And the next router gets it and says, oh, for me that's my interface number two.

And so the packet moves until it gets to the ISP. The ISP then knows not necessarily about anything else on the Internet, but it knows within its own network how to further subdivide that down, just like a letter that is received by the post office in a city. Now, the people in the post office know which carrier to stick it in because they figure out, you know, a given postal carrier has a certain neighborhood that is all these streets. So similarly, once the packet gets into the ISP, it only has to then further segregate it to get it down into the neighborhood, further segregate it to get it down into a smaller block, and ultimately get it to the destination where it's aimed at.

Leo: Again, very slick.

Steve: Yeah, I mean, and the idea that this thing, this system has hung together as well as it has, I'm in awe of the people who did this.

Leo: And no one contemplated, I'm sure, the size and scope of it. I mean, it's exploded beyond anybody's wildest dreams.

Steve: Right. Now, one of the things that has happened is that there are classes of networks. It's worth mentioning. A Class A network is the largest type of network, and that would be all - sort of like the original concept I've talked about where the first byte was a location and the other 24 bits of the 32-bit address were the machine within that location's network. That's a Class A network, where you have basically a network of a single byte at the top, and then 24 bits gives you 16.777 million machines within that. And there are ISPs that own, I mean, many ISPs, actually, own whole big Class A networks. And then they further subdivide that within their infrastructure.

A Class B network has the top two bytes that specify it, and then 16 bits below specify which machine within that network. And the smallest formal type network is a so-called Class C, where the top three bytes specify which network it is, and the lower byte of eight bits specifies which one of 250-plus machines fit within its network. So basically, by having a hierarchy of these network classes, which have actually become a little more loose as the Internet has evolved, we're able to address a packet from one location to another anywhere on the globe. And most of the time, when everything's working right, it gets there.

Leo: Wow. Steve, I think we need to take a break and come back next week.

Steve: Yup, let's do that, Leo.

Leo: I'm digesting. And I don't mean lunch. There's a lot to understand there. Although I have to say it's just - it's really kind of fun to look at it because it's such a beautiful, elegant system.

Steve: Yeah.

Leo: And it works. So, well, we've talked on the show about vulnerabilities, and there certainly are vulnerabilities to this system. But yet day in, day out we use it. We don't think about it. And it works.

Steve: Well, and it's worth also talking about redundancy. I've mentioned that these routers are able to communicate with each other in order to talk about what routes they're able to offer to other routers. One of the beauties of this rich interconnectivity among routers is that, if a router goes down, or it's being rebooted because, you know, its firmware needs to be updated or is being upgraded or something, the beauty of this system is the dialogue between routers allows them to determine when a router is gone. And then they can choose sort of like a not my first choice, but this is the only way I have of sending this packet on. So the Internet is also sort of self-healing.

Leo: There was always the kind of story, although it turns out not to be true, that it was built...

Steve: Right.

Leo: ...to survive an atomic attack. That's not really the case.

Steve: No. That was sort of an apocryphal story that sprung up.

Leo: But I suppose it would because it would route around, you know, big losses like that.

Steve: Yeah. Certainly, though, where our major arteries are located is well known. There's something called MAE West and MAE East, which are major interconnection points. And if bad people, you know, wanted to hurt us, all they have to do is knock out those two major interchanges, and we'd be in trouble. But overall, the architecture, this notion of breaking everything, instead of having wires to everything, just have wires to your neighbors, and send these little packets of data, you know, granulate what you're sending into packets and just drop them on the network and have them all sort of autonomously get themselves to where they're going, it's just - it's a very robust and a fantastic system.

Leo: All right. Well, we'll talk next week about Part 2 of How the Internet Works. And then this is all kind of the foundation for a lot of the security stuff we're going to talk about in weeks to come, I know.

Steve: Right. Because, for example, when we understand how it works, we can better understand what happens when it doesn't work and what attacks are possible to deliberately keep it from working.

Leo: Yes, hmm, all right. We won't tell the terrorists too much. They already know, probably.

Steve: Yeah.

Leo: If you want more information about this, follow up on the show notes at Steve's website, GRC.com/securitynow.htm. Of course, GRC is the great Gibson Research Corporation. Steve's the creator of SpinRite, the ultimate disk recovery and maintenance tool. I wouldn't run a machine without it. I have my SpinRite 6 disk by me at all times, and I encourage you to do the same. And we do thank our friends at America Online for providing us with the bandwidth for this show. We couldn't do it without them. You can listen to the show on their AOL podcast channel or, of course, download it through our RSS feeds, as I'm sure many of you do.

We'll see you next Thursday for another fascinating edition of How the Internet Works on Security Now!. Thanks, Steve.

Steve: My pleasure, Leo.

Leo: See you next time.