SECURITY NOW!

Transcript of Episode #42

NAT Traversal

Description: Steve and Leo delve into the inner workings of NAT routers. They examine the trouble NAT routers present to peer-to-peer networks where users are behind NAT routers that block incoming connections, and they explain how a third-party server can be briefly used to help each router get its packets through to the other, thus allowing them to directly connect.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-042.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-042-lq.mp3

Leo Laporte: Bandwidth for Security Now! is provided by AOL Radio at AOL.com/podcasting.

This is Security Now! with Steve Gibson, Episode 42 for June 1, 2006: NAT Traversal. Security Now! is brought to you by Astaro, makers of the Astaro Security Gateway, on the web at www.astaro.com.

The month of June heralds in graduation, brides, Father's Day, and of course every Thursday in every month is time for Security Now!. Steve Gibson is here to talk about more security issues. Hello, Steve.

Steve Gibson: Hey, Leo. Great to be back.

Leo: Happy June.

Steve: And barbecuing in June, too; right?

Leo: Barbecuing, yeah. June is one of my favorite months, I think because of all of those things.

Steve: Yeah.

Leo: My daughter graduates from eighth grade in a few days. And just the sun is shining, it's summery, I like it.

So you promised us last week that we will talk about getting your router to behave.

Steve: Well, or – yes. What it is about routers which may or may not allow them to behave, and specifically how it is that that security that we've talked about – I mean, you know, we've talked about NAT routers because we're both so bullish on NAT routers from a security standpoint. The fact that a NAT router makes sort of a natural hardware firewall which prevents unsolicited packets, "unsolicited" meaning they're unexpected, you know, the random, as I call it, IBR, Internet Background Radiation. Just this noise, I mean, if you ever put a packet capture on a raw interface, directly on a cable modem or a DSL line or something, it's just amazing how much debris is, for one reason or another, aimed at your IP. Sometimes it's people scanning, you know, for the exploit du jour, whatever it is that has been recently found out. Sometimes it's people still trying to send you messenger pop-ups, spam. It's just incredible how much junk is out there.

So people who have a personal firewall, a software firewall running on their computer, they're typically being harassed all the time until they finally get tired of that and tell their personal software firewall not to tell them every time one of these pieces of junk hits their computer. Actually it's sort of a feature of Windows XP Service Pack 2's built-in firewall is that it's not telling you that, oh, I just blocked something you don't care about. It's silent. And of course all the normal personal firewalls, the traditional Symantec, McAfee, ZoneAlarm, Kerio, Tiny, and so forth, all of those typically have the option of not notifying you every time something comes in. You know, for people for whom a firewall is a new thing, it's sort of fun to say, oh, look, something just hit me. It's like, okay. Stand outside in the rain and you'll have the same phenomenon.

Leo: Yeah. I don't really need anything to hit me, and I don't even know about it if it does. So...

Steve: Exactly. So the beauty of having a NAT router up in front of your computer is that it blocks stuff which is trying to come in your connection which is just from some IP, some random location that you're not expecting anything from. And suddenly your software firewall, if you still had one, that was alerting you all the time, it just goes silent because nothing gets in.

Leo: But the advantage of the router is also its disadvantage when there is incoming traffic that you want.

Steve: Well, exactly. And this has been the great problem, which was – I'm not sure really where the pioneering was done on this. It may well have been the Kazaa people because...

Leo: They had to.

Steve: Well, exactly, because they were running their filesharing clients on people's computers who more and more often were behind NAT routers.

Leo: Now, we should make it clear that incoming traffic's okay. It will always get through if you request it. It's only incoming traffic you didn't request that becomes an issue.

Steve: Well, and then if we're going to say the word "request," we need to explain what that means.

Leo: Okay.

Steve: What a request is, well, I guess there are two ways you could request traffic. You could manually open a port, which is the traditional way of saying I want to run a server. For example, I want to run a web server or, well, I was going to say email server, but that's almost impossible these days because ISPs are blocking port 25, which is what you normally use for email. And even web servers are now being looked askance at by ISPs. But if you wanted to explicitly allow traffic to come into your system, you would need to, in the first case, to open what's called a "static port," that is, create a static port so that packets coming to your IP at that destination port will not be discarded out of hand, which your router would normally do, but instead will be permitted through and sent to a specific computer behind the router in your network which you have designated as the recipient for incoming traffic on that port.

The implicit way of so-called "requesting" traffic is just that you have recently sent your own traffic outwards through the NAT router to some remote location. What that does is that creates a mapping in the router, in the router's RAM – not something that you have to do manually, it's done automatically for you, which is how NAT routing works – so that, when data passes outwards through the NAT router to some remote destination, it implicitly allows return traffic from that remote destination to come back in through the router, and it will automatically be sent back to the computer behind the router that originally sent that outbound. So it essentially allows you to use the Internet seamlessly, connect outwards to any services and servers that you need to use, and anything that they send you will be sent back.

Leo: But as anybody who's tried to put up a web server or an email server or any server of any kind inside his network has found, it's not enough to just have outbound requests. Sometimes you want to have people coming into your system.

Steve: Well, and a perfect instance of this is in a situation where, for example, you wanted to use VoIP or, frankly, if you wanted to just send a file to someone else, and you had some sort of a file transfer protocol. So in the case of, for example, two people who want to have a VoIP connection, what they really want is a connection. That is, we know from our talking about Skype, for example, that, you know, in VoIP the latency between packets, the length of time it takes for the packets to go from one Skype user to another, really affects the call quality and just the feeling of being connected. So what these people want, these two VoIP users, is they want to be able to send data back and forth between themselves.

Well, here's the problem. They're both behind a NAT router. The NAT router will allow data outbound but not inbound without being reconfigured. And what we specifically want to do is we want to avoid that reconfiguration because what that really means is it means opening a port so that traffic can come in from anywhere, not just from the person you're trying to connect to. So, for example, imagine that the first person tries to send some data to the second. Well, that data will leave their NAT router easily. That's no problem. That's what NAT routers do. But the data is going to travel across the Internet to the other person and hit their NAT router and die completely. It's unexpected data. It's unsolicited, so there's no way for it to get in.

Well, what the clever guys, I think probably first at Kazaa, came up with was this notion of both people doing the same thing at the same time. That is, each person sends data at the other person's NAT router. And they send it from the same port that the other person's data came from. So in fact it might stumble a little bit at the beginning, depending upon, you know, whose packet arrives at the other person first. But the beauty is that the outgoing traffic from or through each NAT router creates an expectation of return traffic. If the other person is able to design their outgoing traffic so that it looks like the expected incoming traffic for the other person's NAT router, it'll get through.

Leo: This sounds like hacking.

Steve: Well, I mean, it's some really cool stuff.

Leo: I mean, we've seen people use other systems like a third-party centralized server, like Hamachi users, where the transaction's initiated as an outbound transaction for both ends, and then they kind of get handshaked. But this eliminates – there's no third-party server at all. This is a way to do peer-to-peer transactions without having any third party involved.

Steve: Well, kind of.

Leo: Yeah?

Steve: Actually there's no third-party server involved once the connection is initiated.

Leo: Ah.

Steve: That is, that the third party is not a relay that is relaying all the traffic between the two endpoints. But it is necessary to briefly use a third party, some server, like, located out on the Internet that both of the people who want to connect are able to connect to. The reason is that, until the NAT routers have allowed each other's traffic in, they can't see the traffic that is trying to get in.

Leo: Ah, okay.

Steve: So here's what is really going on. And this is the whole point of what makes a NAT router peer friendly or not, and how it's possible to knit together a conversation between two people both behind NAT routers. Both of their applications, for example, their VoI application, send data from their computer to this third party. We'll call it a "rendezvous" server, or maybe a "liaison" server. There's no real official designation because this is just all kind of stuff that's been created. So both people send data from themselves to this third party. The rendezvous server sees the external port number, and that's the critical information. The parties may know each other's public IP, which is the only way they would have of sending the data to each other in the first place. But normally you cannot know what port your NAT router is going to assign to the outgoing data because that's done just sort of automatically and algorithmically, and it's changing all the time.

Leo: So it's not like web traffic where it always uses port 80. It uses a random port each time?

Steve: Well, see, and that's what NAT does. NAT uses a – it rewrites the packet so that it changes, not only the source IP...

Leo: Ah.

Steve: ...so that instead of coming back to your computer behind the router, it comes back to the router. It also changes the source port so that, when the packet comes back to that port, the router knows, ah, this port is – at the moment it’s assigned to this conversation on this computer behind the NAT router. That’s the whole way NAT is able to disambiguate all the traffic...

Leo: Got it, got it.

Steve: ...coming to it among multiple machines that are all behind NAT.

Leo: But, now, that’s only for internal use. I mean, for external use you’re using the public IP address and the standard canonical port, like port 80. Once it’s past – once it’s outside the router.

Steve: Well, okay. But port 80 would be the destination port. It’s...

Leo: Oh, the return traffic has its own port.

Steve: Exactly.

Leo: Even when you – okay. So this is something I didn’t understand about the ‘Net. So when I’m surfing to a website without a router, I go out and ask for something at port 80, but it will come in on a different port?

Steve: Yes. In fact, that’s where we have this notion of service ports are typically numbered 1 through 1023, and application ports, or client ports, start at 1024 and go up to 65535.

Leo: And they’re assigned randomly in each session.

Steve: Well, actually they tend to be assigned sequentially.

Leo: Sequentially, okay.

Steve: Because there’s really no security problem with allocating them sequentially. But so, for example, as your web browser is making connections to external servers, it’ll ask the OS for a port connection. The OS will normally just assign the next numerical higher port, starting from 1024 and going upwards.

Leo: Got it, okay.

Steve: And then, so the packets that leave will contain that port number as their return. Actually it's the source port because it's the port that sourced the packet.

Leo: Right.

Steve: And then they'll use port 80 as the destination. And so the remote...

Leo: That's true for all protocols. Pretty much.

Steve: Well, I can't make – I can't say that.

Leo: Many. Most.

Steve: For example, DNS. DNS generally sources its packets from port 53 and sends them to 53 so they come back to 53.

Leo: Ah.

Steve: And there are other protocols like, for example, service protocols, like Windows will generally source ports from 445, and they come back to 445. So, no, it's not true for all protocols. But for typical computer client protocols, you know, like web and an email client that is connecting out to port 25 to send mail or to port 110 to receive mail, it'll use high-numbered ports.

Leo: I don't think you can give Kazaa credit for this because I think Microsoft had to create a directory service for its Messenger. ICQ probably had to do something similar, didn't it, for chat? All of these predate Kazaa. This problem must have been solved by them as well, yes?

Steve: Actually, they all had this problem and did not have solutions for it.

Leo: Didn't have a good one.

Steve: Right.

Leo: Well, I know Microsoft used its directory servers at first, and I think that that's how they tried to get around it, yeah.

Steve: So NAT would just normally have shut down those early...

Leo: Couldn't do it.

Steve: ...those early adopters. I mean, and remember that, in the beginning, having a NAT router really was a lot of trouble. I mean, it caused problems...

Leo: Oh, yeah. Oh, yeah.

Steve: ...for people who wanted these kinds of things. Okay, so in order to close this topic, both people send packets to this liaison, or this rendezvous server. It sees what port those packets came from on each NAT router, and it exchanges that information, sending it back to each other, that is, to the other person. And then they subsequently send their data to each other using the proper port number because they've sent data outbound to the other IP, and they've aimed it at the port that now they know, thanks to the third party's intervention, brief intervention, now they know what port the other person's data is coming from. Their packets are regarded as solicited, even though they technically aren't the returning traffic. To each other they look like the returning traffic, and it works.

Leo: And at that point you've got a handshake, and the rendezvous server can get out of the way.

Steve: Yes. And in fact, the other thing the rendezvous server does is it also provides the public IP. Because, I mean, generally, you know, you could look at the email headers; or, you know, each person could check their routers to, like, look at the configuration of the router to see what its current public IP was; or, for example, you could do something like use ShieldsUP! at GRC, which will show you your public-facing IP, even if your own computer is in a private network, because that's what it is that ShieldsUP! is testing. So there are ways you could find out what your public IP is, but they're pretty tech-y and hairy.

So the beauty of this rendezvous server is, when two people send their traffic both to it, it sees the IP from which their traffic came and the port from which their traffic came. That information it swaps and provides to the other party, that then sends their traffic to that port and IP a couple times. And what'll happen is the NAT router sees it, thinks it's expected, and lets it right back through.

Leo: Amazing.

Steve: It's just cool. Now, the problem is – so now we've laid down how it's possible to connect two people, both behind NAT routers. Well, what if a NAT router is hostile to that? That is to say, what this requires – think about it for a second. It requires that the port from which the traffic was sent when you sent data to the rendezvous server will not change, even when data is now sent to a different IP, that is, it is sent to the guy you're trying to connect to, not to the rendezvous server. That is to say, the NAT router has to deliberately re-use that same external port, even when the IP you're sending the data to is different. Not all NAT routers do.

Leo: Oh.

Steve: Mine, for example, doesn't.

Leo: Is that nominally a good thing? I mean, obviously it's a problem here. But is there a reason it doesn't?

Steve: No. And in fact, because this is causing problems, manufacturers are moving towards peer-to-peer-friendly NAT technology. And in fact, even the formal RFC for Network Address Translation, i.e., NAT, suggests that it is better if the public port is kept static, as long as the internal IP and source port remains the same.

Leo: And that's totally for this peer-to-peer sharing.

Steve: It's specifically to enable this kind of peer-compatible operation.

Leo: Interesting.

Steve: So what's happening is we're seeing newer firmware versions of routers are fixing this because their customers are complaining that, you know, they can't get good Skype quality or good whatever because of these problems. So firmware upgrades are subtly changing the NAT logic so that the same outside port will be used. Now, again, this doesn't represent a security flaw because the mapping is still dynamic. That is to say, even though the same port is being used at the moment, the public side port is chosen because the VoIP application is running on a certain machine, and Windows assigned it a certain port which it uses for all of its dialogue. So if it used a different internal port, the NAT router could and would give it a different external port. So it's not like there's any security compromise at all. The NAT is still assigning these port mappings dynamically. And only traffic coming back from an expected port and IP would be able to get back in. But it's just – it's elegant, and it's beautiful. It does require the brief services of this rendezvous server, something out on the 'Net that can see both parties and able to accept the data and exchange it so they're able to connect to each other.

Leo: I'm wondering if there's any way to tell if your router is compatible or not.

Steve: Well, it's funny you ask.

Leo: Huh.

Steve: Because there used to be. But I found out when I was preparing to talk about this, and I wanted to get some URLs handy, that the only service that used to be available that I knew of apparently is gone.

Leo: Oh.

Steve: There was a multiplatform program called NAT Check, and it was a project living over at SourceForge.net, the open source home ground. And it was running on some servers at MIT. And there are apologies now on that page that it's no longer possible to pull up the tables that used to be there. It was, what, it was like last summer that you and I talked about this on Call for Help, Leo, and this NAT Check program worked, and the tables were there. It's all gone. Now, I already have a bunch of stuff on my plate. I'm working on some new...

Leo: We don't want to add anything more.

Steve: ...technology for GRC. I still have to get the OpenVPN docs done that many people are waiting for.

Leo: Yes.

Steve: That's the next thing I'm going to do, as soon as I get this other project finished. But doing a NAT checking service is, I mean, it was made for GRC. It's the kind of thing I should do.

Leo: Yeah.

Steve: So we will talk about it again when I've got that done. But as far as I know at the moment, there is no good way to determine whether your own particular NAT router is peer-to-peer friendly and allows this peer-friendly NAT mapping.

Leo: Can you assume that, if your router is less than a couple of years old, that it will?

Steve: No. In fact, what I remember from that table when it was online was that even recent routers were still not doing that. The only, I mean, this is a horrible, I mean, a burdensome suggestion. But if you had a packet capture capability, that is, if you were running CommView or Ethereal or something and were using Skype – in fact, this is how, Leo, you and I verified that we had a direct connection and were not going through a third party, was I saw that my packets, my VoIP packets, were going directly to you out on the public side of my NAT to your IP, rather than through some third party. So if you had the ability to look at your public traffic, and you were using two NAT routers, you could tell that you had a direct connection just by seeing that the actual IP was that of the other person and not some third party that it was bouncing through.

Leo: And if you don't, you could always check to see if there's new firmware for your router.

Steve: Certainly, if you're seeing this kind of problem, always keeping your router firmware up to date is just always a good policy. But I definitely saw in this original table that they were showing – they had a table, a very comprehensive table of routers by make and model and version of firmware, and you could see the differences in this behavior based on firmware version. So it's certainly something that is tending to migrate towards offering this capability.

Leo: And you might look in the release notes for various firmware releases or your current firmware release for your router to see if it mentions that. How would it describe such a thing?

Steve: Actually, I don't recall, when I was looking at routers back when we were talking about VPN stuff, I was studying routers extensively to understand...

Leo: They don’t tell you, do they.

Steve: ...exactly which features – I never saw anything about this.

Leo: They don’t mention it.

Steve: Which was really annoying. On the other hand, it means there’s really a need for something that will tell people.

Leo: That’s probably why MIT stopped doing its thing, because it was so hard to find the information.

Steve: Yeah.

Leo: Okay. So you won’t really know if you don’t have it unless you’re willing to get a packet sniffer out and kind of hack at the thing. And you can’t...

Steve: It is something that I think I need to do, though, so keep an eye on Security Now!, and...

Leo: And would you notice an improved performance? I mean, would you notice a difference?

Steve: Well, look at the difference we had with Skype.

Leo: Yeah, that’s true. As soon as I started using a dedicated port, it really did make a difference, yeah.

Steve: Yeah. And so there what you were doing is you were compensating for the non-peer-to-peer friendliness of somebody else’s NAT router. See, they both have to be compatible in order for this kind of connection to work.

Leo: Right. Oh, fascinating stuff. Now, how does this relate to NAT traversal?

Steve: Well, this is NAT traversal.

Leo: This is NAT traversal, okay.

Steve: You know, that’s sort of the formal jargon for the notion of how to solve the problem that the statefulness of NAT and the inherent firewallness of NAT brings along. And the idea is you open simultaneous – you do, like, simultaneous connections outbound through each NAT router, having chosen the proper ports to send where the packet is going to be coming from. One interesting thing that has been suggested is that, if NAT routers were logging the incoming denied traffic, and if your applications could each read the log, they would see the packets hitting the outside of the router and be able to adjust their outbound traffic so that suddenly it was expected.

Leo: Would that have the same dangers as Universal Plug and Play?

Steve: It wouldn't, and it's sort of an interesting hack. But it's not something that, as far as I know, has ever been done or is supported now.

Leo: So in other words, this concept of a rendezvous server works on many routers, but not all routers. And those routers it does not work on are not going to – it's going to be difficult to use, or you may not get good quality on some of these services.

Steve: True. And what it really means is that there will always be some third-party service that you're using. For example, you know, Google Talk uses their servers to solve the problem. Skype uses their servers. Peer-to-peer networks use nodes that are open in order to help people find each other. So one way or another you are using a third party in order to provide that brief glue connection that allows each router to get the information from the other one.

Leo: So if you had – if NAT traversal were enabled on both sides, you wouldn't need the third-party route at all.

Steve: Oh, no, you still do because...

Leo: You still do, yeah, okay.

Steve: Because of NAT, they're blind to each other until somebody tells them how to see each other. And then they're able to knit a connection directly through.

Leo: Very interesting stuff.

Steve: It is, really, I mean, it's a cool kludge that has been sort of – it's a consequence of the fact that most routers will behave themselves this way. And it does, you know, it allows you and me to talk as well as we can.

Leo: It's only frustrating because we can't tell our listeners whether they have a compliant router and how to fix it or how to find out.

Steve: At this point, what we've explained to them is why it doesn't work...

Leo: Not how to fix it.

Steve: ...if it doesn't. Exactly.

Leo: Steve Gibson, you're the greatest. Always fun, always fascinating. We will see you next Thursday with another exciting edition.

And of course I do want to remind folks that we have a great sponsor, and we're really happy to have them along for the ride. It's, of course, Astaro Corporation, makers of the Astaro Security Gateway, which is a fantastic piece of software, free for download for home users. You might want to get an old computer and put it on there because it gives you great firewalling. You know, and I will bet that it does NAT traversal pretty darn well. I haven't asked. You know what? I'm going to ask.

Steve: I think it's open source based, and so...

Leo: If it didn't, somebody would have fixed it by now.

Steve: ...I think all of the open source NATs do this correctly.

Leo: Yeah, I'm sure it does. And of course, if you're stuck with Novell BorderManager, you might be interested in ASG because they offer easy migration for Novell BorderManager users. ASG v6.2 features a new single sign-on capability for eDirectory clients, an eDirectory browser, and a generic proxy server. Check out Astaro's BorderManager migration Wiki. It's online at Astaro.com/bordermanager. Or just visit www.astaro.com.

While you're on the Internet, make sure you check out Steve's site, GRC.com. Not only a great place for security information, like ShieldsUP!, the notes to our show, securitynow.htm. Actually we can give you a new address in a second. But also a great place to find SpinRite, which is Steve's day job, in fact, a fantastic program. I've used it for so long. Current version, v6.0, is fantastic.

Steve: Maybe for about 18 years, Leo.

Leo: Yeah, you know, I'm thinking, when is the first time I used it? And it's probably when John and I had you on the radio show, probably – wouldn't be 18 years ago, but it'd probably be '91 or '92.

Steve: Yeah.

Leo: So it's been quite a while. Quite a while. And it's a fantastic program, and check it out if you have a problem with your hard drive or you just want to maintain, make your hard drive, you know, run as efficiently as possible. SpinRite.com.

Steve: Whoops.

Leo: I'm sorry, .info is for the...

Steve: SpinRite.com will take you there, too, but it takes you to the normal SpinRite page.

Leo: Okay.

Steve: SpinRite.info takes you to our testimonials page.

Leo: Got it. And we've always said GRC/securitynow.htm for our show notes, but we don't have to do that anymore.

Steve: Yeah, I added some technology to the server. I noticed that people were not putting .htm, or some were putting .html. And I know, Leo, you have a penchant for short URLs.

Leo: Yes. You know, like my radio show is Leo.am, you know? Nothing more.

Steve: Right.

Leo: And never .htm or .html. So you fixed that?

Steve: Yeah, well, I added the ability for URLs that don't include a file extension to be accepted, and the server fixes them.

Leo: Yay.

Steve: That is, you know, our web server really runs at www.grc.com. So if someone goes to GRC.com, they'll get the www added for them.

Leo: Right.

Steve: And now, if they leave off the .html or .htm, they'll get that added. So you could just do GRC.com/securitynow...

Leo: Hallelujah. How long have I been on you for this?

Steve: Well, let's see, this is Episode No. what, 42?

Leo: Even before then I always – you've been known for your obscure URLs, so I'm glad to hear that. That's great news. And it was an easy thing to do, wasn't it.

Steve: Oh, I'm glad it's done.

Leo: He’s an old-fashioned guy, and that’s what we love about him. Steve, we’ll see you next week for some great information and security news and so forth. It’s a must every Thursday. Make sure you point your podcast client, whether it’s iTunes or whatever you’re using, to Security Now!’s feed, which is Leo.am/podcasts/sn. That’s all you need.

Steve: What is it, we’re about 100,000 listeners now.

Leo: We are, every single week.

Steve: Yup.

Leo: And I think it’s fantastic. Thank you, Steve.

Steve: Always a pleasure, Leo.

Leo: See you next time.