DNS Best Practices

featuring Mark Langston as Interviewed by Will Hayes

Will Hayes: Welcome to the SEI Podcast Series, a production of Carnegie Mellon University’s Software Engineering Institute. The SEI is a federally funded research and development center operated by Carnegie Mellon University and funded by the United States Department of Defense. My name is Will Hayes. I am a principal engineer here at the SEI, and I am pleased today to welcome my colleague, Mark Langston, who will be discussing DNS Best Practices with us. Welcome, Mark.

Mark Langston: Thank you, glad to be here.

Will: Before we begin, could you talk to us a little bit about your background and where you come from, and how you happened into this area?

Mark: Well, as you may know, I work with the Situational Awareness Team as part of CERT under Rachel Kartch, our team lead. I have been in the industry for about 25 years now. I started as a systems administrator. Back then, there weren’t really security-specific roles. A systems administrator was expected to stand up all the servers, stand up all the networking, expected to deal with all the security issues, to respond to security incidents, run the full range of security work. I have spent time as a security developer. I have spent a lot of time doing testing for various DNS software vendors. I was privileged enough to work with some of the people instrumental in developing a lot of the standards for DNS that are in use today. I find in my role here at SEI that a lot of the research I do tends to rely on DNS information as a means to enrich data and help us in the tasks that we have before us.

Will: OK, great. So you came in to this realm of work in much the same way the CERT organization came into being. As I recall when that organization started, it was people who were having to do this as “other duties as assigned” part of their administrative responsibilities relating to systems. They realized that there is a large amount of information that you need to master and to be able to work with, and that information is growing. DNS is probably one of those areas where growth is very prominent, isn’t it?

Mark: It is. When you think of computer security, you often think of things like viruses infecting a computer, and host-based security where you have antivirus software and log analysis and things of that nature. But there is also a great deal of information available on the network connecting the computers to the routers and the switches and the larger internet. That information usually takes the form of information about IP addresses, and protocols, and ports, and the payloads that are contained in the packets going on the wire. Part of that information also comes from DNS. Not only do you see DNS queries on the wire and DNS responses, which can help aid an analyst in doing their job, but when you are working with IP addresses, IP addresses aren't very informative of themselves, which is one of the reasons we created DNS because names are much easier to remember than numbers. So DNS becomes a staple in our toolkit for how we analyze traffic on the network and how we can make determinations about network behavior.

Will: If we think about a layperson's perspective on what DNS is, the example of a telephone book or a switchboard might come into play. If you could help us differentiate why is DNS something more than just a listing of these addresses?

Mark: Sure. The domain name system today, more than ever, is the underpinning of the functioning network that we all rely on. If the DNS system is working great, nobody notices. If it goes down or if there is some problem with it, everything breaks. Your phones break. Your desktops break. All the apps that you rely on break. Websites, apps on websites, they all stop working.

These days, with everybody either partially or fully moving into the cloud, you don't have your infrastructure onsite anymore. It is somewhere remote. If you can't resolve the information that you need to resolve to access that remote infrastructure, your business is dead in the water.

DNS provides a number of services beyond just what everybody thinks of when they think, *I've got a name and I need an IP address that matches it*. For one thing, it does the reverse. It provides a lot of information having to do with security. There are records designed specifically to aid in the prevention of spam and the prevention of phishing. There are any number of uses of DNS that go far beyond the basics, but when the basics break, everything breaks.

Will: This is another one of those items of progress in technology where the fact that it is invisible is one of the things that makes it elegant. The fact that it works without conscious effort is one of the sources of efficiency. By the same token, that fact allows people who have ill intent to remain better cloaked because of that elegant design. Could you elaborate on that?

Mark: It does. You're absolutely right. One of the interesting things that I have seen over time is that many people, and usually for performance reasons, will disable logging on their name servers. It is unfortunate because DNS logging is one of the primary ways you would notice somebody doing something untoward with your DNS infrastructure.

There are other ways, for example network traffic monitoring, looking at traffic volumes that might help you catch denial-of-services attack; that is, leveraging your name system as an unwitting participant against the victim. But you may miss things like cache poisoning where an attacker will go in and name servers, if they don't know the answer to a question, recursive name servers will go out and get the answer and then they save that in a cache until some ticker expires and then it's supposed to be flushed out of the cache. Attackers will sometimes leverage that and go in and change the information that's stored in the cache. So for example, if you're on any typical network today and there's a query for www.google.com, I can guarantee you, it's in your name server's cache. If somebody can go in there and change the IP address associated with www.google.com, you have suddenly got a large problem because now every query for that name is being returned with the attacker's IP address, and they are doing whatever they want to do with those victims.

Will: So people can be going to places they didn't know they were going because the directions…

Mark: Correct and because they trust the domain name system because there is, much like much of the early internet, it was all built on trust. There is never this assumption that people would abuse the trust placed in these authoritative servers. I say authoritative here not in terms of DNS, but just in general. It was assumed that if something was stood up, it is legitimate and it is providing you good information. That's not the case in today's world.

Will: Unlike when you inadvertently dial the wrong number you'll hear a voice on the other end that confirms for you that you dialed the wrong number. In the domain we're talking about, such obvious clues are not always available. In fact, they're rarely available.

Mark: In fact, the attacker is usually on the other line trying to convince you that he is in fact the person that you thought you dialed.

Will: I think there is ample motivation for why this is an important area for research to be done. Could you talk a little bit about what you are currently focused on and the cool stuff you're publishing?

Mark: Well, some of the things that I look at regarding DNS is looking at various aspects of the domain name system that can be used to help enrich indicators of compromise, for example, looking at the age of a domain name registration. The thought there being that domains that have just been registered may be more likely to be used in a malicious manner than domains that have a long and stable history.

I gave a presentation last January, a year ago now, at FLOCON, talking about how bad actors may use the domain name registration system to hide their true identity in such a way that might be identifiable. For example, using storefronts as fake addresses for domain name registration. That is another area that people leverage. When you have an IP address and you can convert it to a domain name, you can then go look up who registered that domain name. By extension, you can take the information about who registered the domain name and go look up all the other domain names they have registered. So, if you know for a fact the domain name is associated with a bad actor, you can go find all the other domain names they have registered and take action against those as well, before they are able to attack you.

Will: That is kind of an illustration of how the elegance of it operating seamlessly in the background can actually serve the good actors as well in uncovering the bad actors.

Mark: Very true, yes.

Will: Neat. Neat. So, what's in the near term horizon? What's your near-term challenge that you really want to tackle?

Mark: Well, I'm starting to do a lot of work with big data. A lot of this information is very voluminous, and querying or searching that information is time-consuming. I am looking for ways to make that much more efficient so we can do more with the data. So, for example, rather than having to wait 30 minutes to an hour for a particular search on an IP to come back, and then another 30 minutes or an hour for a search through the WHOIS registries, and then another several hours to pivot off that onto other domain names and then start the process over again, big data gives us the opportunity to go in and very quickly do all of those things at once. That just gives us an analytical power that we haven't had until recently simply because of computation time being too expensive.

Will: OK. So I understand you have a blog post in the works coming soon?

Mark: Yes.

Will: OK. Is there something in particular people want to look for in that blog post that we haven't yet talked about today?

Mark: Well, the blog post is focused on building a secure, resilient DNS infrastructure. When I say that, what I mean is that the way in which you deploy your DNS servers can impact your security to the extent that you may be the victim of a denial-of-service attack, at which point your DNS infrastructure is not usable, and that means nothing else on your network is usable. You may be a victim of something more nefarious where somebody goes in and changes the information. You may simply be a victim of reconnaissance by an attacker, where if you have not taken the steps to properly secure the information that you are providing to others via DNS, the attacker can use that information to gain information about your network infrastructure as a whole. For example, one problem that I see sometimes is that people have authoritative name servers--name servers that serve the authoritative information for a given domain, say example.com: the authoritative names for example.com is responsible for telling you what the IP address is for www.example.com and mail.example.com and so forth--those are all things that you would want the public to have access to. Some deployments of DNS, however, in those authoritative name servers, they have all the names of their internal machines as well, like hr.example.com and payroll.example.com. These are not publically accessible systems, but the information is available publically. For an attacker, that is invaluable because they can look through that and start to get a sense of what your internal network looks like and potentially what targets they would like to attack.

Will: This really seems to resonate with the principle of least privilege?

Mark: It does, yes.

Will: There's an implication there. Can you elaborate there?

Mark: Yes, definitely. DNS servers—and I've alluded to this several times previously—typically come in two varieties. You have authoritative name servers, whose job is simply to be the authority of information for a given domain or a set of domains, and recursive name servers that aren't authoritative for anything but whose job is to answer any queries sent to it.

One of the things you want to do is make sure, as we just mentioned, that only the information necessary for the parties using the server is available on the server. So, for example, as we mentioned previously, the people outside your organization do not need to use your recursive name server. Their ISP is the one responsible for providing them with name service. So you don't need an external recursive name server. Your authoritative name server does not need to have information about your internal systems when it is an external authoritative name server.

Your internal authoritative name servers—the machines that are used as the gold standard for the information there, which are called primary name servers or master name severs—those should only be accessible to your infrastructure people, your IT folks, and your administrators; in fact, only to those who actually have the authority to go onto your authoritative name servers and make changes to the records on the name server. Nobody should be allowed to query that. That machine or those machines should exist solely to push that information out to the machines that are used by the people internal to your organization or external to your organization. And that's called the hidden master or a hidden primary. That preserves the idea of least privilege, and that only those who need access to it are allowed access, and then only to do the things they need to be doing to it, like patch management and data management and so on and so forth.

Another concept is high availability. Name servers tend to be misunderstood sometimes, and organizations…I remember when I started, the name server was a machine that was pulled out of the closet that they thought, well, rather than throw this away, let’s put it to good use. That served for a very, very small group of people. As organizations get larger and larger, having a single name server either by itself or in a high-availability cluster, an HA cluster, isn’t enough.

For one, everything you do online requires a DNS lookup. The latency between the end user and that name server becomes an issue. When you load a webpage, I don’t even know what the current number is but I know there are tens to hundreds of lookups for any given webpage you load these days because there are so many things embedded in the webpage. There are images that are on different hosts. There are analytics that are being run by third parties. There are all sorts of ads. Those are all separate lookups. Each one of those takes a measurable amount of time, and when your name server is located, say, in your headquarters, and you’re in a branch office 4000 miles away, you don’t want that latency. You want to have those name servers as close to the people who need to use them as possible. You need to make sure that if one goes down, there’s another there to pick up the slack, which would be the concept of the high-availability cluster. That needs to be present throughout your organization. So, you need to make sure your local branch offices have a high-availability cluster. You need to make sure your headquarters have a high-availability cluster for both recursive and authoritative name resolution. Externally, you should have high-availability clusters.

You should also make sure that you have a diverse set of name servers for every name that you are authoritative for. Take the case of example.com. When one goes to register a domain name, you are asked for at least two name servers for that domain name. When deciding what to put in those fields, you should make very sure that it’s diverse enough that should one go down, the other is unaffected.

As viewers may have seen recently, Rachel Kartch, my team lead, did a podcast a month or two ago, having to do with distributed-denial-of-service attacks. She mentioned the then recent incident with Dyn networks. What happened there is that many very large customers of Dyn went offline briefly when Dyn was under a distributed-denial-of-service attack. The reason for that is because they bought name service from Dyn. Although Dyn’s network is very geographically diverse, they have name servers all over the world, they are all still on the same network. So when Dyn’s network was impacted negatively, their customers’ services were impacted negatively. Ideally, you would want to have a name server in one location on one network, and your second name server, your third name server should be on a completely separate network in a completely separate location from a completely separate provider just in case one has an unforeseen event, whether it be manmade or natural. You want to make sure that your services remain accessible to your customers.

Will: If we think about obstacles to implementing best practices, I think the illustration you just gave helps us understand there are cost implications to some best practices.

Mark: Definitely.

Will: The physical separation, the logical separation, least privilege concepts, there are cost consequences to making good choices there. But there may be some more subtle, less cost-driven things, something to do with workloads of individuals, who are responsible for understanding the schema, who are responsible for updating and maintaining currency of the data. Could you talk a little bit about those squishier areas where it makes it difficult to implement best practices?

Mark: That's a very good question. There are several things that could be done. In fact, this being a traditional problem, the entire issue of having to manage a large and diverse infrastructure, there are vendors in the marketplace now that provide solutions that do many of the things I've talked about in one package. They allow you to easily manage a large DNS infrastructure from a central point, and take into account the architectural and security items that we have discussed in this talk.

A good example, there are several vendors that implement DNSSEC very well, DNS Security Extensions. DNSSEC, in a nutshell, is a way to help ensure the integrity of the answer you get when you ask a DNS server a question. The way that happens is through public key infrastructure. There are digital certificates from the root server all the way down to your name server that form a chain of trust between the very top and the lowest end node, in this case your name server, that say to the user who is checking the DNSSEC extensions, *Yes, not only can I give you the answer, but I can guarantee you it is the answer I meant to give you*.

One of the issues with implementing DNSSEC—and it has been a problem for some time now—is the concept of key rollover. To generate the certificates, there are keys associated with them and in order to generate the keys which in turn are used to generate the certificates, there is usually a long arcane process that takes place. To add to that, it has to be done on a somewhat regular basis. I have seen 30 days, 60 days, 90 days, out to a year, but it does have to be done regularly. That can become a bit of a problem for your maintenance staff. Several vendors out there that provide these all-in-one solutions have solutions for that baked into the product that allow you to—once you've set up DNSSEC, which I strongly recommend using—it makes the key management almost invisible, as minimally invasive and as easy to do as possible including things like reminders, *Hey, this is going to happen in the next five days, be aware of it*. *Hey, I've done this you might want to check to make sure everything is OK*.

Will: Correct me if I'm wrong, this is one of those things where when you are trying to navigate to a particular website and you get the prompt, *The certificate has expired. Are you sure you want to go here?* This is one of those the good hygiene practices you are talking about is one the things that prevents that missed connection.

Mark: In that sense, yes. It is similar to that. It is not actually related except to the extent that certificates are used in both cases. This ensures that you actually are able to get to that site, to see that you are in fact going to the site you thought you were going to. This prevents, among other things, cache poisoning attacks where the attacker will go in and change the information because once that information has been tampered with, it is no longer valid in that chain of trust. And it can't be verified when the end user's browser or computer tries to resolve it and then checks those certificates.

Will: So this is another elegant background process with limited visibility. Do you see a potential future exposure?

Mark: Well, with anything new that's introduced, there will always be risks. I'm not aware of any currently with DNSSEC, however, the biggest impediment DNSSEC has right now is simply one of deployment. I know the federal government has mandated that all federal websites use *https*, and that those zones, the names that they're authoritative for, be signed with DNSSEC. I know that the roots are now signed, which was mandatory to start this whole process. You can't have a chain of trust without one end point being signed.

But many organizations, large and small, simply haven't given it any thought; in part, again because it's somewhat opaque to somebody who doesn't understand it. But, in part because beyond the group of people who really concern themselves with the domain name system, and beyond security folks, the word isn't really out there that this is a thing that should be done. You can use IPv6 [internet protocol version 6] as a similar example. People knew for years we were running out IPv4 space. It wasn't until IANA said, *We're not issuing any more IPv4 addresses*, that the industry actually decided we need to start migrating to IPv6. In part, that was because of inertia, and in part, it was because that's not an easy process. Because again, IPv6 wasn't something that was very well-understood a few years ago. And DNSSEC is in a very similar position. I hope it doesn't take a similar circumstance to get the world at large to understand that it's important to ensure the integrity of the information that you're serving to the population at large.

Will: Technology refresh is a challenging issue, especially if there's any, you know, potential for disruption of our current state to refresh.

Mark: Yes.

Will: So let me ask you the kind of the big term or the big picture, long-term audacious goals for work in this area. What would you offer our audience in terms of where you see all of this going?

Mark: Well, my background, it's somewhat odd. I'm actually a trained psychologist.

Will: Wow, OK.

Mark: And I've always found computers and, by extension, computer security fascinating because they are an extension of ourselves in the sense that they reflect how we think about things and how we expect to interact with things. I think there's a lot to be had in terms of behavioral analysis with respect to the domain name system.

The work that I've done recently with WHOIS data is one example. There's nothing requiring them to behave one way or another when they register names. But, for example, bad actors oftentimes tend to want to maximize their efficiency, which is a polite way of saying they like to be lazy a lot of times because why do more work than necessary. So they will use the same registration information for hundreds or thousands of domains. That is a direct reflection of human behavior in the domain name system. I think that that is ripe for all sorts of research in terms of not just thinking about how people use technology but why they're using at that way and then leveraging that knowledge to see if it can benefit us in some ways.

Will: Very interesting because I think there's a way to describe what you just said as the bad actors have an opaque background process that's taken for granted. Once you discover that, you can exploit it.

Mark: Yes.

Will: An interesting note to end on perhaps, so. Any final parting words you'd leave with our audience?

Mark: Well, there is a blog post that's forthcoming that will have more details on all this on the external website. There is also a document that NIST [National Institute of Standards & Technology] puts out that I'd like to recommend. It's NIST SP 800-81-2, which is NIST's recommendations for DNS best practices, some of which we have discussed here today. It goes into more detail than my blog post will about how to properly deploy and maintain a resilient, secure DNS infrastructure.

Will: Great. Mark, I would like to thank for joining us today. This was a very interesting converation.

Mark: Thank you, I enjoyed it.

Will: So, as always, a recording of this podcast is available on the SEI’s website as well as on Carnegie Mellon University’s iTunes U site [and the SEI’s YouTube channel]. Resources mentioned during this conversation as well as other related material will be made available through links on our website. To get to us, you should go to sei.cmu.edu/podcasts. If you have any questions please don’t hesitate to email us at info@sei.cmu.edu. Thank you very much for joining us.