ICANN73 | Virtual Community Forum – DNSSEC and Security Workshop Session (3 of 3)
Wednesday, March 9, 2022 – 16:30 to 17:30 AST

KATHY SCHNITT: Thank you, hello, and welcome to the DNSSEC and Security Workshop part 3 of 3. My name is Kathy and I'm joined by my colleague Kim. And we are the remote participation managers for this session.

Please note this session is being recorded and is governed by the Expected Standards of Behavior. All participants in this session may make comments in the chat. Please use the dropdown menu in the chat pod and select Respond to All Panelists and Attendees. This will allow everyone to view your comment. Please note that private chats are only possible among panelists in the Zoom webinar format. Any message sent by a panelist or a standard attendee to another standard attendee will also be seen by the session host, co-host, and other panelists.

This session includes automated real-time transcription. Please note the transcript is not official or authoritative. To view the real-time transcription, click on the Closed Caption button in the Zoom toolbar. And to ensure transparency of participation in ICANN's multistakeholder model, we ask that you sign into Zoom sessions using your full name—for example, your first name and your last name or surname. You may be removed from the session if you do not sign in using your full name.

And with that, I'm happy to hand the floor over to Russ Mundy.

RUSS MUNDY: Thank you very much, Kathy. And I wanted to start off our session by expressing my thanks and the full program committee thanks to both our ICANN support staff, Kathy who everyone hears. There's other people on from that, also. Kim is there. And Andrew's on. And none of this would be possible without them. So thank you to all of them. And it's a special thanks to our tech support crew, who also are essential to making all these things happen.

And so I am privileged to be able to moderate our third session of the day in our DNSSEC workshop meeting today. And this one is really a panel that is covering several different areas that have some relationship to DNS but a lot of relationship to the security aspects of things.

We have three presentations in this panel. And so I think I want to go ahead and move on into them. And our first presentation is from Max Stucchi, on the RPKI and the importance of it for DNS. So, Max, over to you.

MAX STUCCHI: Hello and good afternoon and good evening to everyone. Let me start sharing my screen same way we just tested a moment ago. And you should all see my presentation now

And let me take a step back, introducing myself. It's my first talk at an ICANN meeting. I used to work at RIPE NCC before I joined the Internet

Society in mid-2019. So I've always been very close but never had the opportunity before to present here.

So what I wanted to present today came from the MANRS Project. And I will get to MANRS in a moment. Before we get there, we need to take a tour to introduce what BGP is. I will be very brief because, similar to what my colleague Robin did earlier today trying to explain quantum computing in 20 minutes. Explaining BGP and RPKI in 20 minutes is also challenge. But I also have some data I'd like to show you towards the end of the presentation that I generated recently.

So BGP is one of the fundamental protocols that makes the internet work. You use it between autonomous systems to exchange information about where resources are, where IP addresses are. It's a very simple protocol but it's very complicated. Works in clear text. Requires collaboration between BGP speakers. And let me start saying some of this might start sounding familiar to you if you are not a routing expert but you're a DNS expert. Because I will make a statement in a moment that I'd like to also ask you to not repeat anywhere else. I know this is being recorded, but I will deny having said that.

BGP works by making announcements. There's a network that says, "Hello, my network, my prefix is this/24," and talks to another network. And the other network says, "My address is this/24." And together they start exchanging data in different directions. But there is a problem.

How do I make sure that the announcement I'm accepting from other networks is true? How do I check that the network that's announcing me something is the rightful holder of that resource? For this reason we have internet routing registries, databases that hold some truth. We'll get to that in a moment.

Basically, parallel to getting my request, my announcement, from Network B if I am Network A, I go to the IRR and ask, "Is Network B supposed to be announcing that network?" And the same thing supposedly should be done by Network B. So we have an external source of truth in a certain way that should give us a definitive answer about who the rightful holder of a resource is.

But we have a problem because some data in these routing registries cannot be fully trusted. Because of accuracy, because of incomplete date, because some data is not maintained. But also, and this was actually, I would say, more true in the past, where not all the regional registries have an internet routing registry. So you have to basically trust external databases that are not run by the IRRs.

And this means that these databases have fewer ways to understand who is the rightful holder of a certain resource. I always use an example. I have my own autonomous system. And in RADB there is an entry for a network of mine that I have not created. And it's there. I'm keeping it as an example. It was true some time ago I was announcing that network from my autonomous system number. But I didn't create that entry. Someone else did it for me.

And this is to reinforce what the problem statement is here. So in some cases there can be no full verification of who holds an IP or an autonomous system number. So here comes RPKI, Resource Public Key Infrastructure. So the idea has been to add a way to tie IP addresses and autonomous system numbers and tie them to public keys. So add a way to verify the information contained in databases. And in one go put this on top of the already existing hierarchy that are the internet routing registries.

So in RPKI everything is handled by the routing registries, by the RIRs, such as ARIN, RIPE NCC, LACNIC, and so on. So this helps us answer the question is a certain autonomous system number authorized to announce a certain prefix. And we have a signature that's telling us this is true. So it was started in 2008 by all the RIRs and provides data that you can trust.

So this works with a framework. I will go a bit quick here because this should already start becoming familiar to you if you are more interested in DNS. It's the same as DNS and then we're adding something similar to DNSSEC on top of it. So please don't quote me on this, but RPKI could be considered as standing for BGP what DNSSEC is for DNS.

We add some ways of verifying the data we have in the registries so that we can influence routing. So we have some pieces of data that go and create my chain of trust. And then we have in this case the RIPE NCC route certificate. And then every member of RIPE NCC in this case gets its own certificate signed by the route private key. And the two can be verified. So you can start verifying the chain of trust.

To this we will add the piece of data in a moment. But another thing that might look familiar to you when you think about DNSSEC is that also here we have two elements. One is the part where you sign something so you create your ROAs. And in a moment I'll tell you what these ROAs are. And when we validate we verify the data that others have put into the system. And the two are separate.

So you could sign something without validating. And vice versa you could validate the data without having signed anything on your part. So the two go separate ways but they contribute, of course, to the final result. That is verify what data we have with BGP.

So what are ROAs? ROAs are pieces of data. You might have heard about route object. I like to say that ROAs are route objects on steroids. Because not only they can be verified in a better way but they also hold a little bit more data. You can create multiple ROAs for one single IP range. They can overlap. You can have multiples ROAs that say that multiple ASNs, different ASNs, are originating the same address space. So you can be very flexible in how you create ROAs and how you sign your resources.

So what is in ROAs? We have a prefix, so the network you want to announce. An origin autonomous system number that tells us who is supposed to be originating the BGP announcement. And then when I say they are on steroids it's because there is an additional piece of data that carries almost my name. That's the max length, maximum prefix length accepted for this ROA. So this means we have a way to say I want to sign. I have a /16 but I want to allow everything from a /16 to a /24. Everything in between is allowed to be announced by this specific autonomous system number.

There are repercussions. There are issues in doing such a big declaration. It's really not suggested to do something like that. But again this is more advanced RPKI, so let's not get into that.

A ROA gets created, gets signed by the LIR resources, which in turn are signed by the RIPE NCC route. And then here we have the full chain of trust. We can verify and validate.

So now that we have the piece of data that shows us what is signed, what we are supposed to be seeing in BGP, we can use it to run validation. Validation is taken from all the RIRs. You might notice that there's a small exception here where ARIN doesn't have a green circle on top of their server there. That's because all the trust anchor locators for all the RIRs are fully available. But for the one from ARIN you still have to confirm you know all the disclaimers.

All the ROAS are normally downloaded by a validator. That's a software that an operator has to run inside their network. And then this validator provides data for the router. So what is the final goal? The final goal is to get ROAs from one side, get BGP announcements from the other, and then use the mix to make better routing decisions. Because now we have, as I said, data we can trust. From the left side of this slide we have ROAs. We have data we can trust that come from the RIRs that I can verify via verifying their chain of trust. And then on the other side I have BGP announcements. And I can see if the two match to at that point make better routing decisions. To be able to dissect what I'm supposed to be seeing in the routing table and what I am not supposed to be seeing.

So some people may be trying to hijack a prefix, to hijack some BGP announcements, or to maybe in the future do also path validation. So my router receives data from the validator. We have the RIR repositories on top. The validator gets the ROAs, verifies them. And then creates a validated cache, which then gets distributed to my router via a protocol called RPKI-RTR.

And this is where the focus should be because—when I get the validated cache—I have a list of ROAs that have the validation for the chain of trust. And then I have to do my BGP validation here. So we have two different layers. From the ROAs we go into the validation of these ROAs. My ROA can be valid, invalid, or it can be nonexistent because they haven't been created yet. When my ROA is valid, I go to the next step.

And this helps me in the BGP validation. We said that the ROA contains origin ASN and a network and the max length. So I can take that piece of data, verify that the BGP announcement matches what is in the ROA. The autonomous system number matches completely the one matching completely the one announcing that prefix. The network is the one that's covered by the ROA. Prefix length is covered by it. Everything's fine. So I can at that point consider the BGP announcement valid.

If any of the two, so the ASN or the prefix length, don't match from the BGP announcement to the ROA, then I have to consider that prefix invalid. Because that means that someone else, maybe another autonomous system number, is announcing the same prefix. There's another one trying to run a hijack or they just made a typo. And so in this way I figure out that the prefix is not supposed to be there. I tag it as invalid. And then I should discard it, not consider it as a valid prefix I should be looking at, I should be having in my routing table.

Now when instead I don't have a ROA or the ROA was invalid from the ROA validation, it means I don't have a way to verify the trustworthiness of the data that I'm looking at in BGP. So in that case the BGP announcement will be marked as unknown for RPKI purposes. Because I don't have any data to try to match it to.

So in a perfect world, we would have 99-point-something percent of the address space covered by ROAs. But my colleague Dan earlier showed that that's not the reality. We are actually at about 35, 40% in coverage at the moment. This happened some time ago. I wanted to check how many of the authoritative servers serving top-level domain data were in networks that were covered by ROAs. And which would be actually valid.

So this was the question I was trying to answer myself. So I built a script that downloads the list of top-level domains and ccTLDs, cycles through them, and then checks all the authoritative servers for them. And from these gets the IP addresses and checks where these IP addresses are. Checks all the BGP announcements, because many of these networks are actually anycasted. So the total is 6,900 nameservers on IPv6 and 7,500 nameservers on IPv4.

And then for every one of these TLDs we checked every BGP announcement, its status. And we picked only the valid and unknown. I left out the invalids which are for another measurement to be run. But here's the situation. In IPv6 we have about a 60/40% for noncovered networks.

Now if I look instead then at the ccTLDs and I go into more details. I divided it into three parts that are: partially covered, not covered, and fully covered. When I say partially covered, it means that there's more than one of the authoritative servers is in a network that is covered by a ROA but not all of them. And fully covered means that all of the authoritative servers stand in a network that's covered by a ROA and it's valid. And then not covered means all of the servers are on a network that doesn't have a ROA covered.

This means that a little less than a quarter of the DNSs are covered by a ROA. 64% have started working towards it. But then there's a good 14-point-something percent that don't have anything.

And if I look at the other TLDs, we are at a 54% of not covered. And then it's a better number for fully covered, a little bit better. But the numbers are roughly the same, as you can see.

So this leads to the last part of the presentation. It's like what could be done at this point. The action that you could be doing is talk to network engineers or network operator that provides you the connectivity service. And ask them to create ROAs for you. Prepare your network also to validate ROAs.

And then you could check if you are able to join MANRS. MANRS is an initiative and a project from Internet Society where there are actions that are promoted in order to have a better presence on the internet.

And one of these actions involves creating ROAs, creating route objects, providing more information so that other networks can validate this information. And you can provide better, cleaner routing for everyone.

And if you have any question, feel free to ask me, Dan, Robin, or other colleagues. But what could the challenges be? In some cases I know that many DNS services are run using legacy address space, which predates the advent of the regional registries. And in some cases this legacy space cannot be covered by ROAs because of issues with the RIRs. So that might be a challenge.

In some cases you might have routers that are not capable of doing validation. Because it requires a certain amount of capacity. Or your network operator is not willing to set up ROV. It happens in some cases. But the more customers ask, the more they're pressured to then do something about it.

And I'm happy to take any questions. I'm right at the 20-minute mark.

RUSS MUNDY: Well, so thank you very much, Alex. And we'll do questions at the end, provided that we have time. And I think we need to move on to Alex's presentation about some ongoing work that she is involved with dealing with implications from email forwarding problems. So, Alex, over to you, please.

ENZE LIU: All right. So thank you very much for the opportunity to be able to speak here. My name is Alex Liu. I'm relatively new to ICANN. I'm a third-year PhD student at UC San Diego.

But today I'll talk to you about some of our recent work on email forwarding titled Forward Pass: On the Security Implications of Email Forwarding Mechanism and Policy. This work is currently under submission, so please keep it confidential.

Our motivation for this paper is a series of spoofed emails received from a UCSD mailing list. As you can see here the spoofed emails look just like legitimate email. And in fact email spoofing has become such a serious problem for this mailing list that our department chair has to clarify that his email is not a phishing attack.

This is why we write this paper. It seeks to understand why the spoofed emails shown here are properly delivered without any warning. And we find that email forwarding is the root cause. More specifically, in a normal case, we would expect emails directly sent from Alice to Bob. And in this case existing anti-spoofing mechanisms work well.

However, when an email is forwarded from Bob to Charlie, existing defense mechanism such as SPF and DMARC can fail to prevent the delivery of spoofed emails. And as we will show later in this presentation and in our paper, email forwarding adds much complexity to existing anti-spoofing mechanisms.

More formally, we identified two problems with email forwarding. On the one hand, email forwarding is never standardized. And some forwarding mechanisms work at odds with existing anti-spoofing mechanisms. On the other hand, anti-spoofing is a multi-party problem. And different parties can make completely different assumptions about each other.

And to show that the problems we identified are real, we conducted a series of measurements and made the following contributions. The first, we document different email forwarding implementations in the wild. We also identified various vulnerable configurations and the implementations in the email forwarding flow. Lastly, we demonstrate attacks that impact tens of thousands of domains and billions of users. We showed that an adversary can easily spoof as important a domain such as state.gov, alipay.com, and facebook.com just by combining the vulnerabilities we identified.

And to help you better understand the rest of the talk, let me start by giving you some background information. One of the widely used anti-spoofing mechanisms is called SPF, which is based on IP addresses.

Imagine subset UCSD want to use SPF. They will start by publishing their own SPF record which specifies the IP addresses allowed to stand on behalf of the UCSD. In a legitimate case, an email from UCSD will be sent from an authorized server. And when Bob gets the email it queries the SPF record of the domain in the Mail From header, which in this case is UCSD. And it thinks the IP address 1.2.3.4 is allowed to send on behalf of UCSD, the email is authenticated.

But in a malicious case, where the adversary does not have access to the authorized server, it won't be able to spoof as UCSD in the Mail From header as the IP address of the malicious server won't be allowed by UCSD's SPF record. So this is how SPF prevents email spoofing.

Another popular anti-spoofing mechanism that is widely used is called DKIM, which is based on public key. For DKIM to work, UCSD would first have to publish a public key in their DNS server. And then UCSD would sign every legitimate email with their private key and attach a DKIM signature header to the email. And when Bob gets the email, it queries the DNS server of the domain specified in the DKIM signature for a public key. Bob then verifies the signature. Here, if the adversary does not have access to the private key used by UCSD, they won't be able to sign the email as UCSD.

Well the last anti-spoofing mechanism we studied here is called DMARC, which is used to authenticate the From header. The header that is visible to users and specifies the sender identity. DMARC builds on top of SPF and DKIM. If either SPF or DKIM passes, it would perform additional alignment test. If an e-mail passes SPF, DMARC checks if the domains in the mail from header are the same. Similarly, as the e-mail passes DKIM, DMARC checks if the domains in the DKIM signature and from the header are the same. DMARC is considered a pass if at least one of the alignment tests is passed.

So here's an example of how alignment tests work. Imagine an e-mail has already passed SPF. The alignment test looks at the domain in both from header and the mail from header and checks if they're the same. In this specific example, they're both ucsd.edu. So the alignment test is passed. Similarly, if an e-mail has already passed DKIM, the alignment test exam is to domain specified in the DKIM signature and the from header. If they're the same, the alignment test is passed.

So everything works great if an e-mail passes. What should we do if the e-mail fails the DMARC test? Here, domain owners can specify a DMARC policy that specifies how to handle those e-mails. More specifically, they have three options: none, quarantine, and reject. If the DMARC policy is none, an e-mail from this domain will likely be accepted even if it fails DMARC. If the DMARC policy is quarantine or reject, an e-mail from this domain, the sales DMARC would either be put into spam or rejected. And in the security research community, we generally consider none to be relaxed policy. We can say to quarantine and reject to be strict policies.

So now we're ready to look at some of the measurements. I'll start by describing how we measure real world forwarding implementations. So we started by setting up a mail server, people send the e-mails. You can register accounts with various services that have forwarding capabilities. Samples include Gmail, Outlook, and Google Groups. You also create accounts with some mail providers such as Gmail and Outlook to receive e-mail.

Once we have everything set up, we send e-mails from our own servers through the forwarding services. We ask these forwarding services, "You forward the e-mails to the receiving accounts we control." We observe the forwarding mechanisms used by each forwarding service. In fact, we observe a variety of forwarding mechanisms used by different forwarding services. For the sake of time, we won't be able to go into the detail.

The high level takeaway here is that each implementation modifies the e-mail headers in slightly different ways. This table is the forwarding mechanisms we observe for other forwarding services. Once again, our measurements show that there's no unified forwarding mechanism used by all services.

Now let's look at some of the vulnerable configurations and implementations we discovered. We use the setup from the previous measurement. In this specific measurement, we create domains that have DMARC policy is none, quarantine, and reject. We send both legitimate and spoofed e-mails from our own domains forwarding services. We ask the services to forward both legitimate and spoofed e-mail to the receiving account. During this process, we observe if the forwarding services implement SPF and DMARC properly. We also observe if there's any restriction on forwarding spoofed e-mail. Lastly, we record whether each forwarded e-mail is delivered or not.

With this measurement, we identified a variety of vulnerabilities with both the forwarding services and the receiving service. We highlight some of the most interesting ones here. For example, we observed that Outlook allows what we call open forwarding. So what open forwarding means is that Outlook allows you to forward an e-mail to any recipient account without verifying that you own that account. Another interesting issue we find here is the ability to override DMARC decision.

So users can force the delivery and forwarding [inaudible] email at Outlook by whitelisting a target domain.

The last problem we highlight here is DMARC none forwarding. Meaning that Google Groups would forward an e-mail from a domain with DMARC none regardless of whether this e-mail passes or fails DMARC. As we show later, this allows an adversary to launder spoofed e-mails through a mailing list such that it looked no different than legit e-mails after being forwarded. And one important you usually find with receiving services is relaxed validation. For example, Gmail would deliver e-mail messages forwarded from known providers even if they failed DMARC.

Now, we'll show you some of the examples how an adversary can successfully perform e-mail spoofing attacks but orchestrating together the vulnerabilities we just highlighted. The first attack we demonstrate here targets any domain that contains Outlook servers IP addresses in the SPF record. The adversary started by setting up their own mail server and creating an account with Outlook. They then send a spoofed e-mail purporting to be state.gov with their Outlook account. Upon receiving the e-mail, Outlook would perform SPF and DMARC check. And of course, the spoofed e-mail will fail both SPF and DMARC. However, here, the adversary can force the forwarding of this spoofed e-mail by whitelisting state.gov at their Outlook account.

After whitelisting, Outlook would happily forward the spoofed e-mail. And now, this Outlook also has the open forwarding issue. The adversary can ask Outlook to forward this e-mail to an arbitrary recipient without the recipient's consent. In this example, the recipient is chancellor@ucsd.edu. For Outlook specific forwarding mechanism, this forwarded e-mail would probably pass SPF as state.gov allows Outlook to send on behalf of it. The spoofed e-mail would also pass DMARC alignment test properly as the domain both the mail from header and from header in state.gov.

In summary, these attacks allow an adversary to spoof at any domain campaigns Outlook server IP addresses in their SPF record and deliver the spoofed e-mail to any recipient. According to our own estimation, it effects 12.4% of Alexa problem in the domains and 38.1% of .gov domains. Example domains that are affected include state.gov, secretservice.gov, mastercard.com, and washingtonpost.com.

Then after that, we demonstrate here targets: Gmail users. The adversary can basically use the same setup from last time. This time, the chance to target alipay.com which does not allow Outlook to send on behalf of it. Once again, they start by sending a spoofed e-mail purporting to be from alipay.com to their own Outlook account. And upon receiving the e-mail, Outlook performs the SPF and DMARC checks. The spoofed e-mail will fail SPF and DMARC check. The adversary once again forces the forwarding of the spoofed e-mail by whitelisting at alipay.com and their Outlook account. This e-mail is then forwarded to chancellor@ucsd.edu. And upon receiving this e-mail, Gmail would perform both SPF and DMARC checks. This e-mail would fail both SPF and DMARC checks because alipay.com does not allow Outlook to send on behalf of it.

However, Gmail was still delivered as e-mail because Gmail recognizes this e-mail was forwarded by Outlook, one of the well-known providers. We termed this behavior relaxed validation. We observed that this attack works for any domain with DMARC none or quarantine. And adversary can perform this attack against Gmail users. Some of the example domains that are affected include alipay.com, mastercard.com, disneyplus.com, hulu.com, etc.

For last attack, we demonstrate here targets: mailing lists that use domains with DMARC none. It allows an adversary to launder spoofed e-mails such that they look no different than legitimate e-mails after forwarding. Here, the adversary needed a server capable of sending spoofed e-mail. They start by crafting a spoofed e-mail and send it to the target mailing list hosted with Google Groups.

In this example, the mailing list is list@ucsd.edu. And upon receiving the e-mail, Google Groups would perform SPF and DMARC test. However, even if this e-mail fails DMARC, Google Group will still forward the e-mail because the spoofed domain eng.ucsd.edu has DMARC none. And for Google Groups specific forwarding mechanisms, this e-mail would probably pass SPF and DMARC after forwarding. And from the recipient's perspective, the spoofed e-mail looks no different than a legitimate e-mail.

Just now, we have demonstrated through attacks that impact major providers and loads of domains. We have disclosed all our findings to all the affected providers. Microsoft, Zoho, and Gaggle.email have confirmed the issues we reported and are actually working on patches.

As for the mitigation, some of the short-term mitigations include disabling forwarding, removing relaxed validation and turning on moderation for mailing lists. However, we do believe long-term mitigations will be more beneficial. This can include trying to standardize forwarding and have more holistic and comprehensive approaches for e-mail security in general.

To sum up, in this work, we examine e-mail security in the context of forwarding. We note that forwarding is complicated as it's never stand-alone. And defending against spoofing in the context of forwarding involves multiple parties and each party can make very different assumptions. We perform a series of empirical measurements that document e-mail forwarding mechanisms in the wild. We identify various vulnerabilities in the e-mail forwarding flow and demonstrate attacks that impact tens of thousands of domains and billions of users. If you have any questions, feel free to e-mail me at the enzeliu@eng.uscd.edu. Thank you very much.

RUSS MUNDY: Great. Thank you very much, Alex. That was very interesting presentation. I understand that there is a problem with getting released on the slides so they will not be on the website. Do you know if they'll be able to be released in the future?

ENZE LIU: Yes. It will be released when the paper is public. Right now, it can only keep it a little bit confidential because we don’t want to interfere with the review in process.

RUSS MUNDY: Great. Thank you. So let’s move on to Moritz. Moritz, go ahead. You have the last presentation of the day.

MORITZ MÜLLER: Thank you.

KATHY SCHNITT: Alex, can you stop sharing, please?

MORITZ MÜLLER: Can you see my slides?

RUSS MUNDY: Right now, we’re seeing the schedule.

MORITZ MÜLLER: I’m sorry. I pushed at the wrong screen. Let’s try again. Is this better?

RUSS MUNDY: It’s a blank white slide. It’s just a slide.

MORITZ MÜLLER: That's the slide. Yes. Okay, great. Thanks. My name is Moritz Müller. I work for SIDN Labs, which is the ccTLD registry of .nl. This is a study that we're currently carrying out together with NLnet Labs, which among others develop software like Unbound and NSD. This study has been carried out for ICANN where we carry out a survey about different DNSSEC deployment metrics, and finally, make an assessment of those. So in a way, it's kind of a full circle. Then you're starting off with DNSSEC measurements and we're now carrying out a survey on how to do measurements in general.

The goal of this project is stated here on this slide. It's basically copied from the request for proposals from ICANN from last year. The goal of this project is to perform a survey of academic and industry literature related to the DNSSEC deployment. Second, to find documents for different techniques and metrics used to measure all aspects of DNS deployment. And finally, to make recommendations to ICANN for which metrics measure to obtain the most comprehensive view of DNSSEC deployment across the Internet.

In this presentation, I would like to share with you our work in progress, what we have done so far, how we have achieved them, some of the problems that we encountered. And finally also, I would like to reach out to the community such that we not only assess metrics that are useful for some parts of ICANN but also to the broader community.

Our approach is probably straightforward. We study carrying out a broad literature study, you could say, of papers published in academia, papers and journals, papers published in conferences. We started off by looking to papers published in measurement conferences in well-known IT security conferences, and then go through these papers, see what kind of metrics they proposed, see what kind of measurement techniques they use, and then finally, also look into the references of these papers to find even more studies which do DNSSEC measurements.

In order to cover the more industry part of DNSSEC research, we looked into presentations at different industry conferences, like RIPE-like workshops like these but also think of DNS-OARC meetings and CENTR workshops, where probably people might present their work but where they might not necessarily write a paper about it so that we can also cover metrics used only in industries so far.

Then we also carry out a gap analysis. With that, we use our own knowledge in DNS and DNSSEC measurements to identify metrics that might have not been covered so far by academia or by industry. And finally, to develop an assessment framework to, first of all, assess the different measurement techniques, how to measure the different metrics. Finally, we can give these recommendations with which techniques you can measure a certain DNSSEC metric and how you could achieve the largest coverage.

People are familiar with DNSSEC and measurements in general. We're already wondering that this is a quite broad scope. This is actually the biggest challenge that we have in this study. So when you think about DNSSEC deployments, you probably first think about on the one side signing the roots, signing TLDs, signing second level domain names or even lower, and you will think of validation is a resolver or not. But of course, there's way more to it.

On the signing side, you could look, for example, at what kind of algorithm they use. Do they use NSEC? Do they NSEC3? Do they rely on DNSSEC automation, where we've heard quite a bit about today? And on the validation side, you might think of things like, what kind of trust anchors do they support? What kind of arguments do they support? Do they support significant signaling mechanisms? And so on and so forth.

Additionally, we have this challenge that things are still being developed, as also we've seen in this workshop today. There's still many more things added to the DNSSEC that we might want to measure in the future and where we might want to have a metric on the future as well.

Then finally, you also have things that are related to DNSSEC. First thing that comes into mind is DANE, for example. DANE would be part of the DNSSEC deployment, potentially, as well.

What we've first done and what we're still currently doing is going through all these studies that have been presented so far and collect all the different metrics that are out there. Of course, there are so many metrics that it's very hard to go through all of them, but we try to categorize them roughly in a few categories. The first is, of course, the resolver matrix, where we look into whether resolvers actually query for DNSSEC-related records, but also whether they actually do validation or not. This is already a fine distinction that we can find that show when it comes to DNSSEC metrics.

Also, then we have whether resolvers—how they handle different validation errors, how they handle transport errors. This is also related to DNSSEC, whether they're able to handle large packets or not. But also other related metrics like whether they use negative trust anchors, for example. That could be a metric as well.

When we look at the domain name metrics, first thing that come to mind and which has been studied by many, many different researchers and industry, is whether DNSSEC records are actually published, but also whether they're actually valid or not.

Also, here you could look into whether a name server, for example, also supports TCP as a fallback mechanism. But you can also look into signature attributes, what kind of algorithms they use, how long are the signatures. And you can also look into operational practices, as also has been presented today already. For example, whether they roll the key, how they roll the key, how often they roll the key, and so forth.

Of course, there's also a bunch of other metrics. For example, they look into more the end user, do they rely on validating resolvers or not? Do they only rely on validating resolvers or do they also have a fallback to non-validating resolvers? Things more related to the DNS software itself, so not what is actually deployed in the wires, but what is available to be deployed. So, does a certain resolver of X supports a certain standard or not? Another broad metric category is whether the DNS ecosystem like a certain registrar supports a certain algorithm or again think of whether a certain registrar supports a certain DNSSEC automation standard.

These are all the different metrics that we found so far and there are, of course, many more out there. But which of these metrics are actually most relevant and how can we measure these metrics such that we can achieve the largest coverage? Here, the measurement technique actually comes into play. The measurement techniques have a large impact on how a certain metric is being collected and how many resolvers or name servers we can cover. Of course, you have all the different kinds of measurement techniques coming from passive measurements to active measurements, whether you're more focused on the end user, the recursive resolver, the authoritative name server. You might rely on a measurement platform like RIPE Atlas or you might rely on some kind of hack where you use, for example, a proxy network, which is not made for creating measurements but still being used for issuing, for example, DNS queries.

So in order to assess all these different kinds of measurement techniques, we are trying to develop assessment framework. This assessment framework has to go to basically look into three different attributes of these measurement techniques. The first attribute, which is probably the most important one, but we still would like to hear your feedback on that, is the coverage. So which part of the ecosystem can be measured with this measurement technique, and is there may be some kind of bias in there?

The second one is reproducibility. This has not exactly been asked for by ICANN. But we believe that if ICANN would like to measure some kind of metric, then reproducibility should be an important aspect to take into account as well. Because I think we want to be sure that the measurement that we're carrying out is well understood and then also increases the trust and the numbers that the measurement creates.

Finally, the feasibility. Of course, if you put a lot of money in, you might be able to increase the coverage but this might come on as a cost. This might be even harder to do if it also relies on third party where you have to convince many, many people to participate, for example.

This is the rough overview from what we have done so far. But of course, we are interested in which metrics are most relevant for the ICANN community in general. For example, which metrics are most relevant when we try to look into the deployment of DNSSEC automation or when we look even further into the potential threat of quantum computers? Also, what is the most important aspect when selecting a certain measurement technique? So is coverage the only thing that counts for you as a community? Or do you think that transparency or any other aspect is relevant for you as well? With that, I would like to thank you for your attention. And then maybe we have still some few minutes for questions.

RUSS MUNDY: Thank you very much, Moritz. We really appreciate the insight into this, I think, relatively fresh work. It's something that we had heard about in the past. Good to know that it's underway. We have just a couple of minutes left for questions for the panelists. I'm not seeing any in the Q&A pod. But if folks have any questions, please raise your hand and our staff will give you the approval to ask your question.

We had some back and forth in the chat room about Alex's presentation and standards organization. So I think this would be really encouraging because I think everyone that was commenting here was cheering on the idea of taking the results of this work into the standards round. Okay. We have Ullrich. Go ahead. Yes, you're already a panelist. Go ahead.

ULRICH WISSE: Well, I would have a question for Max, actually. I wanted to ask if I wanted to check on our name servers, how would I do that?

MAX STUCCHI: That's a very good question. The first tool I would use is RIPEstat, which you can find it stat.ripe.net. You can look for your IP address, you put the IP address you want. Actually, I suggest going in the old UI, not using the new one. I really usually get lost in that. You can put an IP address and it will tell you if it's part of a network that is covered by ROA or not in the main page. So it shows you directly which announcement it is part of, the IP address, and if the network has a ROA. And if it has one, if it validates or not. Actually, the RIPEstart API is what I used for the measurements that they ran.

RUSS MUNDY: Okay. Good. There's Geoff giving us a URL and in the chat room. Thank you, Geoff. Okay. Any more questions for folks on the panel? Well, thank you very much to all three of our presenters. Excellent job, excellent information. Really, really interesting. We hope that folks come back with additional ideas for the ICANN74 DNSSEC and Security Workshop.

Again, thanks to not only the presenters on this panel but all of our presenters today. Again, I reiterate, thanks to the tech staff and all of our support staff for the wonderful job that they've done for us today. And for folks that might have ideas for the next workshop at ICANN74, please keep an eye out for the call for participation, which we'll be coming out probably quite soon because the ICANN74 is coming up fairly quickly on us. Okay. Over to you, Kathy, for closing remarks.

KATHY SCHNITT: Thank you very much, Russ. Thank you for joining us today for the DNSSEC and Security Workshop, all three sessions, one, two, and three. And we look forward to seeing you at the next workshop. I want to thank our fabulous Program Planning Committee, our presenters, panelists, moderators, my colleagues, Kim and Andrew, and of course our techs for making this another successful workshop. And with that, the session has concluded. Please stop the recording.

[END OF TRANSCRIPTION]