ICANN70 | Virtual Community Forum – Tech Day (1 of 4)
Monday, March 22, 2021 – 09:00 to 10:00 EST

KIM CARLSON: Thanks. Can I do this opening real quick? Thank you. Welcome to Tech Day. My name is Kim Carlson. Kathy Schnitt and I will be your remote participation managers for this session. Please note that this session is being recorded and follows the ICANN expected standards of behavior. During the session, questions and comments will be read aloud if submitted within the Q&A pod. I will read them aloud during the time set by the chair or moderator of the session. If you would like to ask a question or make your comment verbally, please raise your hand. When called upon, you will be given permission to unmute your microphone. Kindly unmute at that time and ask your question. All participants in this session may comment 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 comments. Please note private chats are only possible among panelists in the Zoom room. Any messages sent by panelists or standard attendees to another standard attendee will be seen by session hosts, co-hosts, and other panelists. This session includes automated, real-time transcription. Please note the transcript is not official and authoritative. To view the real-time transcription, please click on the "closed caption" button in the Zoom toolbar. With that, I'll hand the floor to Dr. Eberhard Lisse. Thank you.

EBERHARD LISSE: Thank you very much, as usual. My name is Eberhard Lisse. I am the chair of the ccNSO's Technical Working Group, which has been organizing Tech Days for 40-45 times. I have counted it recently and this time, again, it's a virtual Tech Day. As usual, we make some opening remarks, I'll walk you through the agenda, and then we'll start. This time, we have slight changes in the procedures. We have been mandated by ICANN, or ICANN staff, rather, to have mandatory breaks at 3:00 and 4:00 of half an hour each. Personally, I don't think this is helpful because I think it interrupts the proceedings, but because this is a community effort we will poll after each break. We will have a slide for each break that we will run. And then, because the opinion might differ from the first to the second break—it might be that some people say the first break is unnecessary, the second break is necessary, or the other way around—we will poll you using the Zoom feature and we will then take the lesson or the message from there.

First, we have organized this a little bit according to time zones so that … We didn't have any applicants for a presentation from Eastern Europe or Asia Pacific, so we didn't have to accommodate them. The Europeans come first. The European and African time zones come first, and then we'll do the North America and LACNIC presentations. Finally, let's go through the presentation shortly. Giovane Moura from the Netherlands, SIDNL, will talk about DNS monitoring with a new software they have developed called Anteater. Maciej Korczyński will speak about the COMAR project, which is, from what I understand, a way of detecting whether registrations are going to be used for purposes.

We'll then hear from Ulrich Wisser after the break about the NSEC3 to NSEC Transition of .NU run at the Internet Foundation in Sweden. Then, Mark Robertshaw from Oxford Information Labs will talk about the RDAP Implementation at Netistrar. Then, we're going to have Benno Overeinder from NLnet labs and IETF. [inaudible] speak about a toolset they developed for DNS Key Ceremonies. Then, Iliya Bazlyankov will speak about ccTLD security practices. And then, we will have the other break, and then two Brians, Brian King and Brian Lonergan. Brian King from MarkMonitor and Brian Lonergan from Donuts will speak about "Homoglyph Domain Names." I was tempted to put a homoglyph in the title, but I must say I thought better of it, to explain what is, but I leave it to the two Brians.

Then, Jorge Hernández will give the usual host presentation, as we all know. We always offer the host of the meeting, usually the ccTLD manager or whoever else is hosting, an opportunity to speak about a topic of their choice with a little bit of emphasis on the operations of the entity behind it. Jorge was so nice to be available. We asked him last year in Cancún, which then was cancelled, so we asked him this time again and he was readily available. And then, we will have a mini workshop led by Champika Wijayatunga from ICANN about setting up email address internationalization. And then, as usual, we have closing remarks, [inaudible], from the Czech ccTLD manager. He will basically give his take on all presentations and what he found good, what was helpful. If we start, then, with the first presentation from Giovane Moura, Giovane, you have the floor.

GIOVANE MOURA: Yeah, thanks very much. So, somebody is going to be passing my slides, I believe. All right. Yeah, good afternoon, everybody, depending on where you're from. My name is Giovane. I'm going to be presenting this monitoring tool we did here at SIDN labs together with colleagues at USC ISI. Next slide, please. So, this talk today is actually based on a technical report we wrote. You can see the link, here. We discuss how we can actually use, if you're running an authoritative name server yourself or you hire third-party services, you can actually prospect and analyze a TCP DNS queries that come to your server to actually measure latency.

So, in this report, it's very detailed, we show how rich DNS over TCP is. I mean, it provides latency measurements and how you can actually use that for Anycast engineering. We actually give a full presentation of that at DNS OARC. So, in this presentation here, the idea is to give the core concepts and do a little demo if it's possible here. But you have the link here for YouTube, where you can see the full video. And today, it's mostly focus on the tool, [introduce the tool.] We actually open-sourced the tool right after the DNS OARC presentation. Folks asked that, so participants at OARC 24 asked whether it was open source. And I said, "Well, we never did it, but we could," and we did later, and that's the goal this talk here too. Next, please.

So, we all know here if you're an operator of an authoritative DNS server, you really are interested in latency, measuring latency, and you want it to leave the fastest response that it can. But it's kind of hard to actually measure that because there are different ways … I mean, operators have different tools to measure that. Usually you can use

RIPE Atlas to measure latency from clients to the authoritative servers … Can actually use private services like ThousandEyes. You can use Verfploeter but it's a little tricky because it requires ICMP and it has some issues with BGP so it's kind of complex to actually map the entire client population. It's hard to use Verfploeter. It's a tool my colleagues have developed with IPv6. So, even though operators will use multiple Anycast, multiple name servers, multiple use Anycast, different peerings, it's hard for them actually to know latency, what the clients experience. Next.

So, we wonder if there is a better way to measure latency and measure a method that actually comes from real clients, that actually works well with IPv6, that requires no extra measurements. It's basically passive measurements only. And while there is—next—DNS over TCP, you can actually analyze the TCP queries that come [to you already run] your server. You can measure the RTT, the round-trip time, between your client and your server during your handshake to [inaudible] session establishment. And we have been using that at SIDN for [over and a half] years, and helped solve several issues you can actually see in the paper, and fulfils all the requirements involved. Next.

So, if you look at the history of TCP RTT, it's old but it's gold. It has been used since 1996, I believe is the year that I got my first computer, which is a pretty long time. It's widely used for HTTP. I mean, Facebook uses it. There's a paper here, number five here that documents it. And a lot of people actually came up with this idea multiple times. In our case, it was Casey Deccio who gave us the idea. I was working with Casey on a

different project and they later applied this idea to this particular project. Next, please.

So, what's new with our contribution? In this paper, we do an extensive and comprehensive evaluation of the methodology. We look if the data is representative, we compare performance with TCP, and we act upon the data with four operators, Anycast operators A, B, B-root, one of the root DNS servers at Google. We identify several use-cases and issues. We use BGP to fix it and we document it carefully. We use that in near real time within .nl. This is the link for Anteater which is the tool we're going to be talking today. And if you are interested, just if you don't want to learn Anteater, if you just want to see the RTTs on your packets, my colleagues at ISI, they have instrumented a tool they called dnsanon. I'm going to post later in the chat. Wes, maybe if you are here, I think you're here, if you have the link, could you maybe please post in the chat, for this tool which allows you to extract, now, TCP RTTs for PCAP files? Next, please.

So, requirements for TCP RTT, TCP traffic must provide enough coverage in comparison to [UDP] traffic. You want to see the most clients, and it has to have similar latencies to UDP, so you can generalize the results. Next slide, please.

So the question is, is DNS TCP traffic representative? And what I want you to focus here—and this is green lines here that shows the ratio of TCP traffic in comparison to UDP, it's on the Anycast A and B, first two columns, but second and third columns, it ranges from 2.6% to up to 6% of all the queries are TCP, but when you look at the resolvers, third

and fourth column, you actually—fourth and fifth columns, you actually see that the number of resolvers where these queries are coming from is like 21% of the resolvers, 23 send queries, and half, roughly 45% of the [inaudible] send TCP queries. So it depends on how you [inaudible], but 5% of the clients coming from 20% of the resolvers and 44% of the [inaudible] send TCP. And it's all for free. You can measure latency for all these folks. Next, please.

And you want also to be sure that the latency to TCP, UDP is the same so you can generalize the results. This figure just shows the CDF of latency towards L-root, one of the root servers, and the lines are really close together for both UDP and TCP, median and the 90th percentile, which shows if you have a latency of UDP measurement towards a certain IP address, you can assume that the IP address over UDP would experience a similar latency to that too. Next.

So, what can you do with that? So, as you're going to see in the paper, TCP provides enough vantage points to get [inaudible] clients that really matter. It has similar rates to UDP. You measure your real clients and not like a population of vantage points distributed on the Internet. Has pretty much no cost, it's passive data, and it copes with IPv6 if you have already deployed it and requires no extra measurement and can be run in near real time, you just need to get data, the PCAP from your servers and process it. Next.

So this figure here, it's also in the paper, it shows one of the .nl authoritative nameservers and on the X axis here, you see these different acronyms and each abbreviation here is an airport code for

one of the Anycast sites of .nl for this particular server. And what you should focus here is the green bars, like these whiskers. The ones that are very far high like LAX A on the figure above, it shows that the clients that go to the LA Anycast site experience a latency from, I don't know, 40 milliseconds, you can see on the right axis, Y axis, to up to 160, and Anteater allows you to see this kind of information and allows you to see, hey, there's something up to this site, so let's see who's actually getting there, similar for IPv6 which is below, the Charles De Gaulle—that's Paris—site so you can actually have an overview of the latency that each site within your Anycast network has delivered to its clients. Next slide, please.

And this is Anycast B server for .nl over IPv6 queries. So on the figure on the left, we have different autonomous system numbers, the first one is Google, it sets most of the queries to .nl and I think this is—I'm not sure exactly what data period is that. It's in the paper from one week of data if I'm not mistaken, or Google has sent, I think, one day. Again, sorry, we have to look it up in the paper, but like the general idea holds is that Google sends most of the queries, and you see the RTT from Google is around 40 milliseconds, but AS 4134, it's pretty high, looks like they send a lot of queries, still not as much as Google, but their RTT is above 250, as you can see on the figure on the left, the green color there if you match both of them. Next slide, please.

So you can actually have this analysis. And one of the problems that you can solve with this sort of analysis is what you call distant land problems. It's when a client is mapped to an Anycast which is very far geographically from the client location. So in this case, in this figure

here, for example, some slides have a large RTT like Charles De Gaulle or Singapore or Tokyo Narita here, these are the green arrow bars that you can see here have a large spread and you can actually figure this out using TCP RTT. Next slide, please.

And one of the examples we saw, it's the Narita site, as in Japan, and based on these measurements from Anteater, we went there and figured out what was going on, and when it pulled all the data from ENTRADA—that's our data warehouse—that was arriving at the particular Narita site for Anycast, most of the autonomous systems that it came from are in China, so it had had to jump from China to Japan and China—I never measured China myself, there have been some scientific papers that show that China occasionally can have international connections exhibit a congestion. So that would be a lot of more tricky way to fix that to reduce latency to other sites in China. One way to fix this would be to have a site in China, but many clients of this operator would not be comfortable with that are directly peer with Chinese ISPs and I've been told that it costs a lot of money too. But this is start of an analysis we can do by analyzing the RTTs that your clients experience. Next.

Another problem we can also see, we actually found this using Anteater, is there's a policy on BGP, it's like preferred customer, and it's a common BGP policy, which means that if [internal] systems can satisfy the route of their customer, so be it, just go there. But sometimes it can take clients to another continent. So we thought, for example, that clients from Comcast which is US-based are actually reaching one of the Anycast B sites for .nl on the Brazilian side, and, well, they're

geographically far, which means they're going to have a higher latency, and this is the left part of this figure below here, you see the RTT, those bars, they vary from, I don't know, 25 up to 150 milliseconds. So we talked to the operators to say, hey, this shouldn't be happening. There are many sites in North America [inaudible]. We think it would be best if Comcast could actually be served by sites in the US or Canada. And they fixed that, and you can see that on the 23rd onwards, latency is reduced significantly. So this is the sort of analysis that even without running your own Anycast, you can actually just go to PCAPs and tell your operator to have a look at that and they fix it. And they have and they did, by the way, and we are very thankful to them. So you can see just by analyzing this passive data, we can actually spot problems and ask them to fix stuff. Next slide, please.

Another problem we found which we defined as Anycast polarization, we found that Google and Microsoft, these [autonomous] system numbers in parentheses, had a very high latency towards Anycast A in the network of .nl, and they're among the top two clients of this particular Anycast network. And you can see the latency of Microsoft is around 80 millisecond and Google varies from five up to roughly 100. Next slide, please.

And it turned out that both Google and Microsoft were only seeing the entire Anycast network as one unicast network. What does it mean? The entire autonomous systems were only reaching one site. So we just see in this phase here in the left part of the graph which is written "announced," that means that these are the number of queries from Google and where they went, they went to the purple line which is like

[inaudible] I guess. They went out to Amsterdam site. So doesn't matter what were the .nl queries that Google wanted to know, even though this Anycast sort of had six sites, they all went to Amsterdam. Didn't matter if they came from Asia or somewhere very far. They all went to Amsterdam. Next slide, please.

And then you see the latency being very high. you can see the red color here. That median latency for our sites were 100 milliseconds. So we did a bunch of BGP manipulations and talked to folks from Google—thanks, Warren, for that, and then later [inaudible] and what matters is sort of the phase which is written "turning," which is the rightmost part of the field, it's kind of a purple color. You see that the red line, before, was going from 100, it's now going around 10 milliseconds. So by identifying this issue, this polarization and fixing it, figuring out that Google doesn't have to be polarized, it can go to all of our sites and therefore clients in Asia can be served by sites in Asia, clients in North America by a site in North America and Europe, actually reduced the median latency for all of them. Next slide, please.

So, that's the sort of analysis we can do. Anteater is just a tool that we did when we released it as an open source tool which is based on ENTRADA. ENTRADA is just another tool that my colleauges—Maarten actually at SIDN developed that tool which ingests PCAP files coming from authoritative nameservers. You can see by the arrows that are exported to ENTRADA, ENTRADA convert them to another format which allows us to easily query using SQL format.

And then later, [we just export this data,] so we get this data from ENTRADA, aggregate it and provide, compute statistics for the time periods that we were interested and then [inaudible] database and that is connected later to a graph on a dashboard and then operators at .nl, researchers can have a look at that. Next slide, please.

So this is a URL for Anteater. You can check there. It's open source. I didn't talk to anyone from ICANN if I could do a screenshare here, because I have a demo. Is this possible? Can I just try here?

KIM CARLSON: Yeah, one moment and we'll promote you.

GIOVANE MOURA: Thanks. Because at OARC, people really wanted to see that, so I thought of, let's just do this. So I'm a cohost now so I can share my screen. So this is actually what we have in .nl. This is the last seven days of data. This is the number of queries that we get per hour for ns3.dns.nl for IPv4 for example, NS1, and here's the RTT that we have per hour for each of them. And you see that all of them, per server, they're usually around 50 milliseconds.

There was a spike here, they went above 66. So for all these graphs here, you can get all this information for free, and if you see the historical here, it can go back as long as you want it, and you can configure alerts within [Graphon] and say, hey, if something goes above 80, just notify me. So this is one dashboard we have. You can also see the unique

resolvers each server gets, unique autonomous systems. So this is one dashboard.

We also have a dashboard that shows different Anycast sites, so one Anycast like ns3.dns.nl, for example here, has many sites, as you can see here, and this is the time [series of] queries, this furthest graph here. You can see that they have different number of queries per hour. This is per hour. And here below, you see the IPv6 RTT. Again, you see here for example that [London]—that's the yellow color on top—is getting 133, so I should investigate what's going on with [London] in this case here. And you can see all the different data points here. And I also plot resolvers and the RTT of IPv4 here. You see the load is pretty good in IPv4, it's 20 milliseconds, but IPv6, it's getting much higher.

So again, this is all for free, you can configure alerts, and this is for one of the servers. You can configure a panel like this for each of them. And Anteater has some scripts that can actually generate this automatically for you. So you just have to configure a little file and import the entire dashboard into [Graphon] later.

And the last dashboard that I wanted to share with you folks here is what we call Hyper Giants, [autonomous systems if you're interested] to know more about them. So we all know that DNS means traffic is heavily concentrated. Wes and I and a bunch of colleagues did a paper showing that for .nl and [inaudible] and Sebastian Castro there too, 30% of the traffic comes from five cloud computers, and Facebook too included.

So we wanted to know, those folks who send a lot of [inaudible], if they're getting short latencies. We were very interested in that. So here, we have like another figure, [time series,] for different [autonomous] systems as you can see here, Google [inaudible] IPv4. This is all IPv4. So you see AWS, [what kind of a] TCP RTT experience on the particular time, and see evolutions. Anything that goes above, we can configure alerts and we can look back later to see what happens. And the same is for v6 here for these hypergiants. In this case here, Cloudflare has experienced 129 milliseconds, the red color, and I'm going to investigate later what's going on with v6. And you can also see how many Anycast sites each of them are reaching as well.

So this all comes for free from the PCAP data, and that's pretty much how we've been using that for over one and a half years now. So we're interested in not only providing statistics but provide data that can allow an operator to identify issues with their own Anycast infrastructure and then work on—actionable data, that's the word. I'm going to stop sharing and go back to the slides. Next slide, please.

And I think that's it. DNS RTT, TCP RTT are very useful for Anycast engineering. We show a lot of use cases in the paper, we've been using [Anteater for one and a half years now.] You can get the tool, and my colleauges at ISI have deployed these calculations over TCP as well with dnsanon, but Anteater can be freely downloaded here. That's it. Do you have any questions?

EBERHARD LISSE: Thank you very much. Just had to find the unmute button again. We're not at all strapped for time. We have these breaks in-between, so if we overrun into the break, it's not a problem. Are there any questions in the question and answer pod? No open questions I see here, which means—

KIM CARLSON: Eberhard, there's one hand raised from Mohamed El Bashir.

EBERHARD LISSE: Mohamed, you have the floor.

MOHAMED EL BASHIR: My apologies, it was a raised hand by mistake. Thank you.

EBERHARD LISSE: Okay, then we have Muhammad Altaf. Also by mistake. Anyway, I would rather prefer if you post your questions in the Q&A pod on the bottom. That's easier, because then we will not overlook it, because the Q&A pod is being monitored by staff and they're very good at what they do.

all right, Giovane, thank you very much. Well over my head, I must say, but I just run a small ccTLD. But it's good to see that serious stuff is being developed, and it's even better to see that it is open source so that medium to bigger ccTLDs, also maybe gTLDs, new gTLDs can look at this to increase their service quality. Thank you very much, and then we will switch to Maciej Korczynski.

MACIEJ KORCZYNSKI: Hello everyone. My name is Maciej Korczynski. I'm an associate professor in Grenoble Alps University, and today, I will present the COMAR system that we developed in Grenoble Alps University that is a machine learning approach capable of classifying compromised versus maliciously registered domains. And this work is funded and in close collaboration with SIDN labs, so registry operator of .nl domains, and also AFNIC, registry operator of .fr domains.

So here is a brief plan of my presentation, and let's start with the overview of DNS reputation systems. DNS reputation systems can detect malicious domains using different techniques and also at different, let's say, phases of lifecycle of domain names. So one group is capable of detecting malicious domains at the time of registration, so example is the predator system or the system also deployed in EURid.

The second group of systems is the reputation system that can detect malicious domains during the domain activity phase, and it's mainly based on the analysis of DNS traffic.

Now, those systems, they can classify domains as either malicious or benign, but they do not consider compromised domains. Now, why do we even need to consider compromised domains?

So compromised domains are domains that were registered by a benign user but got hacked. So as a consequence, they also have legitimate traffic that we may not want to block. So the mitigation action for compromised domains is different from malicious domains.

So things that we need to consider is, should we block, take down, completely remove the domain from the zone file, or perhaps we should notify the hosting provider itself?

Now let's take a look at a few examples. On the top, you see the URL that is used to perform a phishing attack and to collect PayPal information of the victims. Here you also see a screenshot of the malicious page. Now, the question is, what can be done to mitigate this abuse?

To answer this question, we need to take a closer look and investigate a little bit. So on the right hand, you can see the content of the second-level domain name, and here we see that there is no meaningful content. When we take a look at the submission date, it was ... The phishing URL was submitted on the 15th of January 2021 and when we compare it also with the WHOIS information, then we can see that it was registered just two days before the actual URL was blacklisted.

So we can see that this is technical abuse. This is maliciously registered domain name, but there is also website content abuse because there is illegal, abusive content being shared using the URL and the domain name itself.

Now, the appropriate action can be block, take down the domain name, completely delist it from the zone file, but also clean the hosting content. And the intermediary that could mitigate the problem is DNS service operator and also hosting provider.

Now, let's take a look at another example. Here is the URL again that is used in phishing attacks against PayPal, and the same question, what can be done to mitigate this abuse. Now, let's take a closer look, and here we see that the content of the second-level domain name itself is completely legitimate. And it corresponds perfectly also to the name of the domain name and the creation date was in 2014, the registrar registration expiration date is on 2022.

So it's a website content abuse. The domain is perfectly benign, but the website itself is compromised. When we also take a look at the malicious URL itself, we see here "wp-includes." That's a strong indicator that the domain was actually hacked using vulnerable WordPress content management system, perhaps some WordPress plugin.

Now, what should be the action taken? Definitely, we cannot block the domain name, because that might cause collateral damage and prevent legitimate users from accessing legitimate content hosted on this domain name. So the appropriate action would be to clean the hosting content.

Now the intermediary, again, it is not the DNS service operator. We cannot delist legitimate domain name. But who should act? It's more hosting provider, and perhaps also the end user, so the registrant or the operator of the website depending if the hosting is managed or unmanaged.

Here, I would like to also present very briefly the website structure of compromised and malicious domain name. So on the left hand, we see a compromised website, we see the home page in black, and malicious page in blue, and we see a lot of internal links to both HTML pages but also non-HTML content. So as you can see here, by delisting the domain name or suspending the domain name itself, we might actually cause a lot of harm to legitimate businesses, for example, and also the users that would like to visit and access the content from that domain name.

On the other hand, on the right side, we see a maliciously registered domain name where the appropriate action should be actually block and delist the domain name itself. So coming back to our motivation, we really need to consider distinguishing compromised from maliciously registered domain names, because the mitigation action would be different. But this is not the only motivation of this work.

Now, distinguishing between compromised and maliciously registered domains can also lead to creating more effective domain blacklist feed and therefore also give us better insights into attackers' behavior. So now the question is, do current URL blacklists give us the correct insights?

So once more, let's take a look at the example. So here, we see a graph from the phishing activity trends report from APWG working group from the third quarter of 2020. And what we see here is the percentage of phishing attacks that are hosted in HTTPS.

Now, what we can see in the report is that 80% of the phishing sites have SSL encryption enabled to fool victims. We also can read that 8.3% were organization validation certificates. Now, what does it really mean? How to interpret those results? Does it mean that the attackers maliciously registered domains and put their SSL certificates, or perhaps it's legitimate user that used more and more TLS certificates. And also, what does it mean those 8.6%? Does it mean that the attackers actually go through a painful process of getting organization validation certificates? That sounds unlikely.

So for those reasons, we really need to distinguish between compromised and maliciously registered domains and that can really give us insights into attackers' behavior. And it's not only about deployment of HTTPS but also about, for example, preferences of the attackers in terms of pricing and so on and so forth.

So, very briefly, our contributions. So we developed the COMAR system that is machine learning approach to classify domains exhibiting malicious behavior as either compromised or maliciously registered. Also, the idea of the COMAR system is to use really publicly available datasets. So for the purpose of the research, we also used for example [IP DNS] provided to us by Farsight Security. But even if we exclude this data set, we achieve very high accuracy of around 97% with only 2.5% of those positives.

So we developed 38 features to identify the state of the domain name, 14 of them are completely new. We also introduced methods to estimate missing values, in particular the domain creation time in case

there is no access to WHOIS information. And domain age is one of the 38 developed features. We also show that the content-based features are the most important ones in representing domain status.

Now, here is the overview of the COMAR system. So, as an input for the COMAR system, we take URLs that are reported and blacklisted. The next step is we excluded all the URLs that do not contain domain names that are all based on IP addresses. Also, domains from subdomain providers, free ones, also dynamic DNS. And then for the remaining URLs—and in particular, domain names—we collect different types of datasets, like WHOIS information, screenshots, host information, content, technology used to build the website, sitemap and so on.

And then from those collections of the data, we extract features, and those are passive and active DNS, content-based features, lexical features of the URL, and in particular, domain name, popularity features, host-based features, certificate features.

And then based on precalculated models, we automatically classify the underlying domain name of malicious URL as either compromised or maliciously registered together with the confidence interval and also the list of features that contributed to the decision the most.

Now, in total, as mentioned before, we implemented 38 features in seven different categories. Also, we categorize them because sometimes, for example, registry operators might not be able to collect some sets of features. So we proposed lexical features, ranking system and popularity features. So for example, the number of times it was in

the Internet Archive, Alexa, Majestic, Umbrella. And note here that for example in our previous research, we show that it is relatively easy for the attacker to manipulate Alexa list, Alexa 1 Million. But here, what is important is that ranking popularity based on Alexa is only one of 38 features. So even if the attacker makes this effort to put its maliciously registered domain to Alexa 1 Million, it still will be most probably not enough to manipulate the outputs of the classifier.

We also used passive DNS from DNSDB, but we also removed it from the operational, let's say, implementation of the classifier. We kept it for the research purposes. We also collect content-based features, so the number of internal, external hyperlinks, the content length, vulnerable technologies and the number of technologies in general. Here, the idea is that legitimate users will invest more time and effort to serve meaningful content to their customers and visitors. Existence of homepage, using redirection techniques and so on.

Also, another group of features is WHOIS and TLD-related-based features, things like Spamhaus index, domain name age. Also TLS certificate features and active DNS features.

What if we cannot collect data for some features? This is true that in practice, there are always missing values when it comes to collecting data. And if we cannot collect the data, we cannot classify domains, or even worse, we might misclassify them and for example, suspend a legitimate domain that was misclassified as maliciously registered. So we need to fill the missing values correctly.

So we proposed in the paper a few methods to, for example, fill the missing value when there is no registration date to estimate the age of the domain name or to replace the missing content features, because it is not always possible to fetch the content, if we, for example, see that the malicious website is armed with Google reCAPTCHA, which we actually see more and more in phishing attacks.

I'm not going to go into the details, but this is just to encourage you to take a look at how we did it in our research paper. But just to mention that for example, for registration date, if there is no WHOIS information, then we try to take a look at different datasets like Internet Archive, certificates, [inaudible] logs and so on, and then based on the available data, estimate the registration time also.

In our method, regarding the dataset, we manually label very carefully 2300 domains, and they represent phishing and malware distribution URLs from Anti-Phishing Working Group, OpenPhish, PhishTank and URLhaus. We use two machine learning approaches: logistic regression and random forest. Random forest, it does not require a feature transformation. As you will see, it performs a bit better in terms of accountability. But the advantage also of logistic regression is that it's a parametric method and thanks to this, we can also interpret the results more easily and see, for example, which features are more important than others.

So now, here are the results. So in the table, you can see the results, the accuracy, precision, recall, F1, MCC. For random forest, for logistic

regression, but also for a few heuristics that we managed also to implement that were used previously in global phishing surveys.

So the domain, according to the method used in global phishing surveys, the domain is believed to be malicious if it is reported, very short time after registration.

Unfortunately, here, there is no more information what is the actual value there, so we applied three months. So if it's blacklisted within three months of the registration, it is considered as malicious, contains brand name or misleading string, and we did not implement the registration in batch because of a few reasons.

But in general, what we can see is that for random forest, we achieve a very high 97% of accuracy, whereas for the method that is used in global phishing surveys for those few heuristics that we managed to implement, it achieves also quite high accuracy of 85%.

Now, when we take a look at false positives—and false positive is a situation when maliciously registered domain is classified as compromised. And in this case, COMAR only has 2.5% of false postiives whereas for Anti-Phishing Working Group, we observed 26%. And it might be because, for example, the attackers decided to age the domain name and they simply started using them in the attacks after a few months period.

Now let's take a look at also risks of false negatives, so the situation when the compromised domain is classified as maliciously registered. And here, we particularly took a look at the domain aging problem, but

also the main goal here was to take a look at only compromised domains.

So what we did here is we took 10,000 domains from completely independent dataset, not manually labeled by us. Those 10,000 domains, we found them in hacking forums and so on, and they were really hacked. We've seen the evidence. And we checked how fast they were actually hacked after the registration.

And what we could see there is that as many as 12% of the domains—this is quite a lot—are compromised within first three months after the registration. So that really may lead to increased numbers of false negatives. So it's not only maliciously registered domain names that are being aged and they stay really long dormant before being used in the attacks, but it's also the compromise domains that get hacked very soon after being registered.

So a few more results. Based on our manually labeled dataset, for phishing domains, we've seen that there are 58% maliciously registered and 42% of compromised domains. For the sample of malware domains, 57 are compromised and 43 were maliciously registered, but now we also are applying COMAR to unlabeled data, and here, the results might change a little bit.

Now, what we also evaluated in the study was the importance of each group of features. So what we did, we were applying logistic regression on the phishing and malware datasets by removing one feature set at a time. So here, you see the percentage in terms of there's accuracy,

precision and so on, so the higher the better, and also error. So false positives and false negatives. So here, we expect to have a lower, low values.

And what is interesting here is that those results show that when removing the content-based features, then the accuracy goes significantly down in comparison—especially for malware dataset, and at the same time, the number of errors—here for example for malware dataset—is more than 30%. So that proves that really content-based features are the most important.

Now, we would like to also briefly discuss the phishing landscape 2020 report which provided a lot of very good insights. It was a very good study on the scope of disruption of phishing. Unfortunately, there are a few things about COMAR that were incorrect, so we would like to just briefly precise a few things.

So according to phishing landscape 2020 report, the approach here to distinguish between maliciously registered and compromised domain is similar to the one that is used in global phishing reports. So the domain is believed and labeled as maliciously registered if it appeared in the blacklist within seven days of being registered, contains famous brand name or misleading string, or there is some indicator of malicious registrations in batch.

So we could read in the report that the approach, at its core, similar to the COMAR methodology which was designed by researchers at two security-minded ccTLD operators. So in fact, the heuristics that were

used in the phishing [landscape in the core] are completely different, because here, we propose really 38 different features and we do not make any [inaudible] assumptions about the features. For example, setting up the domain age to 7 or 30 days or saying that if the domain is listed in Alexa 1 million, it means that it is actually compromised.

What is important also is that we extensively evaluated this machine learning approach. That is completely also automated. So also in the report, we could read that in one way, the method is more conservative than the COMAR method which considers a domain to be maliciously registered if it appeared on the blacklist within three months of its registration time or if it has a famous brand name in the string in the domain name.

So once more, those features are just only a few out of 38. And as mentioned before, we do not make any assumptions about, for example, the time when it was registered, so the domain that was hacked just after the registration, there are very high chances that it will be still classified automatically with accuracy and high confidence level as compromised.

So just to conclude, COMAR leverages publicly available datasets and makes classification decision based on the extracted features, 38 features. The system can be used by different intermediaries to decide on appropriate mitigation actions.

It can serve as an effective tool for creating domain blacklists from the existing URL ones. So after applying COMAR, the domains labeled as

maliciously registered could be, for example, blocked by ISPs at their local resolvers.

And what is also very important, we discussed extensively evasion methods, how we could trick the COMAR classifier, and it's relatively hard and it would require, from the attacker, quite a lot of effort to actually evade the system. And I encourage you also to take a look at our paper.

So, very brief acknowledgements. Acknowledgments of course to SIDN and AFNIC for great collaboration and funding the project, the data providers, Anti-Phishing Working Group, OpenPhish, PhishTank, URLhaus, Farsight Security, and also other projects that supported our efforts. And thank you for your attention. Just drop a line if you would like to discuss a bit more the project. Thank you.

EBERHARD LISSE: Maciej, very nice presentation. I have a few remarks and questions from the chair until I see a hand or a question in the pod. My question is, aren't you mixing up domain names versus websites? Malicious domains and compromised websites. It's probably not a big difference, but should one not be more exact?

Secondly, I have a bit of a problem going into content on a principle basis. However, if this is machine learning-based, that you can flag the ones that you must bring to human attention, that's probably acceptable.

And the last thing is, is this tool available in some other form, preferably open source?

MACIEJ KORCZYNSKI: Yes, so we are discussing this if here would be a possibility to open source it, but we do not have decision yet from the steering committee of the project. This is something that we'll definitely discuss and consider.

EBERHARD LISSE: And then the question with regards to conflating domain names with websites.

MACIEJ KORCZYNSKI: I'm not sure if I understood the question. Perhaps you could repeat.

EBERHARD LISSE: Are you not conflating—you're saying compromised domain name and maliciously registered domain name. I think what you mean is compromised website versus—

MACIEJ KORCZYNSKI: Absolutely. Yeah.

EBERHARD LISSE: And one should make this point a little bit clearer because never mind that it's not easy to just take down a domain name in some jurisdictions, it's very difficult—and I am very reluctant—to decide based on [consent.] 38 parameters is very good because the more parameters, the more refined this is. But in any case, if you can use this tool to sort of reflect this is a suspicious domain name, bring it to human attention, then I can live with this.

There is a very interesting question in the pod from Yoshiro Yoneya which I like very much. Are you using or going to use RDAP instead of WHOIS?

MACIEJ KORCZYNSKI: Coming back to your questions, yes, absolutely. What we mean is the compromised websites, because of course, the domain itself could also be compromised by for example domain shadowing attacks. But here, the great majority of compromised websites aren't [inaudible]. So yes, it's about maliciously registered domains and compromised websites and actually benign domain. So this is to answer your first part of the question.

And second part, yes, now we are working with SIDN and AFNIC, and the idea is actually to provide the information to a person who could evaluate the automated decision made by COMAR. And it's not that COMAR only gives as an output a label, let's say compromised or maliciously registered, but it also gives the list of indicators why the classifier thinks so, why the classifier actually gave this particular label

and also the confidence of the classifier itself. So yes, this is something that can support for example helpdesks in registry operators or registrars or hosting providers.

And now the question was, are we going to use RDAP? We could use RDAP, but also, the number of domains is not that significant. But it would be implementing RDAP, it wouldn't be a problem. Of course, it would even help much more in the project. But for now, we didn't have problem with collecting and parsing the key fields in WHOIS information because also the number of domains is not that high, but definitely, if we have many more domains and we experience some problems with collecting WHOIS, then absolutely, RDAP is the next step.

EBERHARD LISSE: Why not do both? If RDAP works, leave it there, if not, use WHOIS. Future proof it.

MACIEJ KORCZYNSKI: Yeah.

EBERHARD LISSE: Next question, from Kristof Tuyvteleers from .pe, are these domains compromised, hijacked, or is the infrastructure that makes use of the domain names to be reached compromised? Can you see that with your tool?

MACIEJ KORCZYNSKI: Could you please repeat?

EBERHARD LISSE: Are these domain names hijacked, or is the infrastructure hijacked? Can you see whether the names or whether the infrastructure is hijacked?

MACIEJ KORCZYNSKI: So it's not a DNS technical abuse. The great majority of the hacked domains are simply because of the vulnerable software. So definitely, it's not DNS infrastructure abuse. I hope that answers the question.

EBERHARD LISSE: Rubens Kuhl from Brazil says, "We are seeing domains where the attacker changes the DNS servers, keep the resource records pointing to the same servers as before but adds a new malicious subdomain. How would your method react to such?"

MACIEJ KORCZYNSKI: I don't know what would COMAR say, but there are strong indications that a domain is maliciously registered, and that would be probably the output of the classification. That said, there might be corner cases. For example, as mentioned before, it could be domain shadow attack where it's not the website that is compromised but it's the domain name that is compromised and just credentials of the registrant were phished, or for example, zone poisoning attack. But generally speaking, my feeling is that those domains are maliciously registered and then it's

just the attacker who controls the entire DNS infrastructure and changes the resource records regularly.

EBERHARD LISSE: Rubens followed this up. He meant the domain is not maliciously registered but a subdomain is. That's difficult to assess anyway because subdomains are not often published in WHOIS and so on.

MACIEJ KORCZYNSKI: Yes. I see two possible attack vectors. One attack vector is really domain shadowing where the attackers really phish for credentials of the registrants and they do not, let's say, change for example the mapping between the domain name itself and the hosting infrastructure but rather, they add subdomains and those subdomains are extensively used in, for example, phishing attacks.

Another possibility would b e zone poisoning attack. This is something that we measured in the past where we can—if the nameserver of the domain name supports nonsecure dynamic DNS updates, then the attacker could simply add a subdomain and point it to the hosting infrastructure related, for example, to phishing. And that would be definitely technical abuse but still, the domain name itself is legitimate.

EBERHARD LISSE: The last question that we're taking before we go on our break is from Rod Rasmussen. Can you talk about confidence levels in the classifications that COMAR provides? In other words, can you classify a

domain with certain features more accurately than others? And what does this do to the analysis of some substance of both malicious and compromised domains?

MACIEJ KORCZYNSKI: So, yes, definitely. This is also the reason why we are using logistic regression, because logistic regression—unfortunately, I don't see the question from Rod anymore, but yes, we provide confidence levels each time, so as mentioned before, the person for example who verifies it can, based on the provided automatic confidence level, decide if the domain is really maliciously registered, compromised or maybe there is false positive or negative, and the second part of the question was, I think, related to ...

EBERHARD LISSE: If you have the question and answer pod, click on "answered."

MACIEJ KORCZYNSKI: Okay. I cannot display it anymore. But the second part of the question was about the features. So the second part from Rod, could you perhaps repeat, please?

EBERHARD LISSE: Can you classify domain research and features more accurately than others? And what does that do to the analysis for some subsets of both malicious and compromised domains?

MACIEJ KORCZYNSKI: Okay. I'm not sure still if I understood it, but generally, once more thanks to logistic regression, we can see also the weights related to each feature and therefore, we can really see which feature, let's say, contributed the most to the final decision. And the other part of the question, perhaps we could take it offline with Rod to perhaps discuss it a little bit more so that we understand the question.

EBERHARD LISSE: Thank you very much.

MACIEJ KORCZYNSKI: Thank you so much.

EBERHARD LISSE: I really would like to see research like this go into open source or into the public domain so that these tools can be used by the wider audience, if only to add it onto current projects to refine the outcome. Thank you very much again.

MACIEJ KORCZYNSKI: Thank you so much.

EBERHARD LISSE: We have a small break until half past. Kim, can you please put up the slide part one? So this one is staying for about 20 minutes until the break is finished, and then we will put a Zoom poll on that you can then please answer.

All right, see each other and talk to each other again in 20 minutes.

[END OF TRANSCRIPTION]