ABU DHABI – ICANN GDD: Monitoring System API (MoSAPI) for registries
Thursday, November 2, 2017 – 09:00 to 10:15 GST
ICANN60 | Abu Dhabi, United Arab Emirates

FRANCISCO ARIAS: Hello everyone. This is Francisco Arias from ICANN org. This is the session on the ICANN Monitoring System API. So we're going to start. This is the agenda for this session. But let's start with the first part.

So a little background on the gTLD space, there is a service level agreement in the registry agreement for gTLDs that defines 12 service level requirements for DNS, RDDS, and EPP in terms of availability, response time, and update time. They are there. I'm not going to be talking much about them. Perhaps the only -- the monitoring system that we have, it's monitoring only a few of them. Basically, focused on the elements that are in the next slide.

So besides the compliance services that I mentioned before, there is also another table in the registry agreement that talks about emergency thresholds. These are thresholds that are beyond the service level requirements in the previous table, and if service gets to those levels, then ICANN has the ability to request or invoke something that is called Emergency Back-End Registry Operator and take over the operation of the TLD.

Most of the monitoring at the moment is focused on the emergency threshold, so many of the things you will see in the API that I'm going to describe are based on that.

So what is the SLA monitoring system that we have? It's based on a monitoring platform that is open source, available for anyone to use. The company that develops that platform is Zabbix. That's the base that we use, plus some custom code that we contracted this company to develop for us. All the code is available in their website.

The SLA Monitoring System, or SLAM as we like to call it, also contains some other parts that are developed internally, in-house in ICANN. The system consists of approximately 40 probe nodes that are distributed globally. Part of the registry agreement will focus on the areas where most users of the services are. So we are changing those from time to time. The system is also comprised of a couple of centralized servers that compile, analyze, and act on the data that is collected from the probe nodes. So for example, that's where the emergency thresholds and the service level thresholds are calculated and [inaudible] are three year, etcetera, etcetera.

We also have a Network Operations Center that operates 24/7. They are the first line of communication with the registries in case of an issue. They call them, they respond to their calls, and

if needed, they escalate things to a team within ICANN that also has an on-call role that is also 24/7.

This is just a graphic depiction of the -- simplified depiction of the system with the different probe nodes and the centralized nodes.

So we have the SLA Monitoring System and what we are talking about here is an API that we have been making available to the registries for quite some time now in pilot mode. And we are preparing to release an updated interface and also in a few months to release it -- that's our production service.

So this API service is taking the information that we have and captured by SLA Monitoring System and make it available as our REST API for registries to use. Just recently, we made this available to ccTLDs also. So if a ccTLD is interested in getting access to their data, we provide this to you.

Even though the monitoring is based on the SLA for gTLDs, we monitor all the TLDs and the root zone. Of course, we don't have any SLA established with -- ICANN doesn't have an SLA established with ccTLDs or the root zone operators. We do that part, the monitoring, as part of our function to maintain the [inaudible] of the DNS.

The benefits of this monitoring system API is that it allows the registries to see, in close to real time, the data that we are seeing. So you are seeing the same thing we are seeing. One of the things we have seen through this, what is it, four years since the launch of the new TLDs, the first delegation of the new TLDs, is that when we were seeing issues and we were talking with the registries affected by that issue, often times we would tell that registry we are seeing this thing and they would say, "No, I'm not seeing that."

So there had to be some dialog to get to [inaudible] where we would say, "Okay, we are seeing the same thing. This is what is happening." So in order to improve that situation, we are offering this API that, by the way, was requested by registries a long time ago.

We are also incorporating a functionality that has been requested by some registries. This is only focused on gTLDs. This, of course, does not apply to ccTLDs, since we don't have an SLA and we don't call the ccTLDs when there is an issue. So there would be the functionality to have a maintenance window that registries, the gTLDs, would be able to set and say to ICANN, "From this period of time, from here to here, this service is going to be in maintenance." So with that, ICANN will stop Emergency Escalation services.

Now we only intend to do that for the first threshold of those services, which is 10%. We send alerts for 10, 25, 50, 75, and 100% of the emergency threshold. So in the case of RDDS, that means we are going to be -- what is it, 10% is 2.5 hours, something like that. If a registry, if a gTLD defines a maintenance window, then the first alert that will be provided would be at the 25%, which is 6 hours. So hopefully that will give a good amount of time for maintenance windows to be conducted. And from what we have seen, usually maintenance windows for RDDS are less than that. As I said before, this will be provided to gTLD and ccTLD registry operators.

Okay. So let's start now talking about what are the methods that are going to be available. We are introducing session handling. That's something we didn't have in the past. Every comment had to be authenticated. Now we are going to be using a cookie to authenticate the request, and there will be a requirement in the system to re-authenticate, if memory serves, every hour. But of course, that's configurable.

Throughout this presentation we're going to be referring to the base URL. It's really not important, but this is the internet URL that we are going to be using. There will be a version in the base URL so that we can identify later versions of the API when that is needed to change something. And of course, the TLD has to be

there since all the information is only provided to the -- well, it provides information to the registry about their own TLD.

So the first method that you will expect there is, of course, a login where you are going to provide your credentials, a username and password. And there is also what listing of IP addresses. You're going to provide IP addresses that you intend to use for accessing this service. These are the same IP addresses that are used in the case of the gTLDs for the RRI system. The reporting interface to which you provide to ICANN on monthly reports, and also the daily escrow reports.

The potential results are: you login successful, or you have invalid credentials, or you are trying to login from an IP address that is not in the list that you provided to us. This list, by the way, in the case of the TLDs provided through the GD portal, in the case of the ccTLDs is a manual process or a radical process through email.

There is of course a logout method. I don't think I need to explain too much of this one.

As I said before, we are introducing, with this new version, the use of cookies, so you will need to provide the proper IDN that will be the expiration time for this cookie.

So let's talk now about the monitoring methods that are going to be provided in that API. But before that, a couple of definitions. In the system we have what we call incidents. An incident is created in the system when a number of sets of testing, or rounds of testing find the service down, four consecutive sets of testing find the service down. In the case of DNS, for example, we test every minute. So every probe node since according to each of the IP addresses of the name service of all the TLDs every minute. And the value here is 3. So if three consecutive minutes we find the DNS service down, an incident will be created for the DNS service of a given TLD.

In the case of RDDS, it's two. So every two consecutive periods; if there are two consecutive periods of downtime, we create an incident. I should mention that the frequency of monitoring for RDDS is five minutes, not one minute. That's the reason why we have this value of consecutive measurements in two for RDDS.

An Emergency Threshold Alert can be caused by one or more incidents. So you can have an incident, for example, last half an hour and then you have another incident later, not continuous, that lasts for another half an hour, then you get a total downtime of an hour. And then you have an alert of 25% for DNS, for example. That would create an Emergency Threshold Alert.

One of the things that we need to remember, as I mentioned in a slide before, the emergency thresholds are based on a seven-day continuous calendar window. So the last seven days, whatever they are, not a week, but the last seven days from the period of time in which you are now, that's what is counted for the emergency threshold. And I had a slide that show exactly that. So that's the rolling week, as we call it.

So the first method to monitor is a general method to monitor the state of a TLD. I'm not going to get into details in the left of the screen about the output. The important thing to know here is that you will be able to see here the status of each of the TLD services. So, for example, for DNS you can see here the status set as down, the percentage of emergency threshold which, at the moment, you are inputting this method, and the incidents that are part of that emergency threshold, as I mentioned before.

An emergency threshold can contain multiple incidents, one or more incidents. And you can see at the top, there is a status for output outside of each of the services. That will be marked as down if any of the monitor services are marked as down. That's just to give a quick overview to the registry that is looking at that.

Perhaps another important point there, there will a "LastUpdateAPIDatabase" that's the field that indicates when was last this information updated. So, for example, suppose something was wrong with the SLA monitoring system and the API is still working and you are able to see the information, you will see that, for example, the information was last updated an hour ago. So you have an indication that something is wrong with the system itself and not necessarily with your service.

In regards to the incidents, there are a few other points that are over there. There will be an ID that is created by the system, and there will be a start time of the incident, an end time. There is a field that indicates where the incident has been marked as a false positive. We have in the system the ability to indicate that something that the system is seeing as wrong, we mark as a false positive and so it would not be counted as downtime for you. But you will still be able to see that information. And, of course, you can also see the state of the incident, which can be active or resolved. Hopefully, those are self-explanatory.

If the incident is still active in the end time, you will mark as "null." It still has no end date. Like in the example here.

Another method that is available is to monitor the alarm status of a service. So in the previous one was for the TLD. You're querying the TLD and it will give you the summary of the status

of the TLD, all the services in the TLD. This one is for a specific service. In example here we're getting, querying, the status of the DNS. You can see there in the example URL that is there. It will give you a simple response where it considered down or not. Or if it's disabled, it will tell you that. For example, for the ccTLDs we have the monitoring of the RDDS disabled, so you would see that. For gTLDs, at the moment, we are not monitoring EPP, so EPP will always show this for now, as "monitoring disabled."

Another method available is for you to query the downtime of the service. It's just the minutes of downtime of the service during a given rolling week period.

You can also query for the list of incidents in a given period of time. We intend to limit this to a window of 31 days. To be clear, I'm not talking about the last 31 days. It's just that you will only be able see 31 days in whatever period of time we have available. We intend to keep all the information available, so you can query the history. But this method will only return you the last 31 days of the information. So you provide a start date, an end date, and there are some defaults if you don't provide one of those. They are described in the API. I don't intend to go into details. There is also an option of a search parameter that is where the incident has been marked as false positive or not, in

case you want to only query for the incidents that have not been marked as false positive or the other way around.

You can drill down and go to see the specific incident with the incident ID that you get from methods, like the previous one. You can use that and query the state of a given incident. And you will see the status together with some other details about the specific incident.

You can also query the false positive flag of a given incident. This is the only field that can change after an incident has been marked as resolved. So even if an incident was marked as resolved we, in ICANN, can come back in later and identify that something was a false positive. So we'll mark that as such. So you will have the ability to check that.

Another concept that I sort of talked about before is measurements. So we talked about incidents and I said that for an incident, for example, for DNS, there would have to be two sets of measurements. As a matter of fact, that's the concept here. Two sets of measurements that are considering the service down in order to trigger the creation of an incident. Measurement, we are going to be talking about. We provide the ability to query the measurements for a given service.

With this method you can get all the measurements for a given incident, and you're going to get lists. And with the next method, you can query the specific measurement. So remember that a measurement is the results for a given probe node. You're seeing the results from a given probe node and a given service, in case you wanted to drill down on specifics and you wanted to get information from where we are seeing the issues in a given service. So you can get an idea if this is a localized issue that we are seeing or if this is a more global issue that is affecting the service from the point of view of the system.

This is just the rest of the details. I don't intend to go into that here.

We provide some error codes. These are the error codes that we provide for DNS/DNSSEC at the moment. Yes, I know these are very limited and we intend to improve the situation. Four is just not enough by all means. It's in the roadmap of the system, the SLA monitoring system, and by sanction, in the Monitoring System API, to provide way more detail on this regard. But, at the internal launch of the Monitoring System API early next year, we are only going to have these methods for now. But, for now, we in ICANN are, of course, available to provide those details. We have been providing the logs of what we are seeing to the registries when they ask us for that.

We also have a set of error codes for RDDS. Again, they are very limited and we need to provide a little bit more detail, and we intend to do so in the near future.

So that's for the methods related to monitoring that are available for ccTLDs and gTLDs. I mentioned before, we are also introducing something that was requested by some gTLDs, it's the ability to define a maintenance window during which we are going to suspend the sending of alerts to the registry. But as I mentioned before, we only intend to offer this, at least for now, suspend the alerts for the 10% of the threshold.

And remember, this only applies to RDDS, and when we start monitoring EPP, also to EPP. We don't intend to suppress any alerts for DNS at the moment. This of course is subject to discussion with the registries. If there is interest, we, of course, are willing to talk about it and consider it, but for the moment this is what we intend to offer.

So you, the gTLD registries, are going to create these objects in the system. You're going to create a maintenance window or a schedule object, as we call it. And you're going to set up a descriptive name for the maintenance window, where the maintenance window is enabled or not. You can disable them, given maintenance windows that you created. An optional description of the maintenance window and the start and end

time of the maintenance window. Here's an example of the schedule object.

Another important point here is that per the registry agreement, we are asking that the start time of the maintenance window is at least 24 hours ahead from the current date and time. The length of the maintenance window cannot be greater than the service level requirement for the given service.

So this would be then create or update for a maintenance window. You just need to put http request, and either create, if the object did not exist, or update, if the object already exists. And you are the one that sets the ID for the maintenance window. We are requiring a standard syntax for the ID. This is the UUID is standard in the ITF. I don't remember what is the RFC, but there is an RFC that defines that syntax for that ID. You will not be able to update a maintenance window for which the end time is in the past. You can also delete a scheduled maintenance window if you wish to do so.

You can list the maintenance windows that have not ended for a given service. You will receive the IDs, and these IDs were the ones you created, then you can query those IDs to get the details on the maintenance windows, as we saw before.

So this is it in terms of the maintenance window functionality. One other thing that we are introducing here, as of now, we provide the list of probe nodes that are used for monitoring. And that's currently in a URL that is open to the public. We intend to change that, and make it only available through the system. So only the registries, those that have credentials to the system, would be able to see this in the future, when we launch the system. This is probably something that will affect the operation of your systems, so you may want to take note of that.

Requesting access for gTLDs, it will be through the GDD portal. We're using the gTLD portal, as this is the general method we have to identify you. That's the place where we know we're talking with you. So we are relying on that to establish the process. Once you have established the credentials, then you just use the API and you don't need to use the portal for that anymore.

Now, in regards to the ccTLDs, we don't have something like the portal in which you know we're talking to the right person, so we have been using, in the current pilot, and we intend to continue doing that when we launch this new version, to use the contacts that are listed in IANA.

So we basically send an email to the administrative and the technical contact with a random string, just like IANA does, we're

mirroring the process they use to obtain confirmation from the contacts that the person that is requesting the access to this information is authorized by the two contacts to do so. And then we create credentials for them in the system, and they get access to the monitoring system API. In order to request access, there is the email address. Except we have a pilot ongoing, and you can already request access. The interface of the current pilot is likely different to what I just presented.

When is this going to be available? Unfortunately, at the moment, I don't have a date. I was hoping when I scheduled this meeting that I would have a date ready. We are pretty close to finishing the project. The best estimate I have right now is that it will early next year. As soon as we have a committed date for the launch of the system we are going to, of course, notify you. Yes, Maxim? [AUDIO BREAK]

As a matter of fact, that was the end of the presentation. [AUDIO BREAK]

MAXIM ALZOBA: Maxim Alzoba for the record. Just a short list of questions. The first: Actually, in our contract we have only one method of communication with ICANN, formal. So can we send it like as an

official note to the contract saying that, "Yes, hereby, we request access to the system."

The reason to ask is that since you accept more important things via this channel, like carbon copy via mail service. Because you accept even change to the contract this way. And it's less important, actually, it's just monitoring. Because the reason to ask is we cannot use -- some of registries cannot use the current portal, because the click through agreement was added in front of the portal, which is not acceptable.

Okay then, the question about the slide about sessions. What is the reason to limit it to two sessions? Do you think that you're saving resources? Because I don't see that it's going to consume a lot. We, as you have more than one monitoring system usually in place, and those who have administrative and technical contact, in case of emergency all the systems are going to access monitoring, and it's more than four. Because we have an administrative contact who is charge, and he usually likes to see what is going on. Technical contact, who is like senior engineer, or something. And they're going to try to login at the same time. And at least two are monitoring the system, which directs it from different locations. So please make it four or five better. It will not damage the system, I think. What do you think about it?

FRANCISCO ARIAS: This is Francisco Arias from ICANN org. Yes, I mean, the number, we had to choose a number and two seemed like a good idea. Maybe not, given what you just said. It's just a matter of resources, as you put it. So we'll look into that. So to be clear, you are saying four?

MAXIM ALZOBA: Five.

FRANCISCO ARIAS: Five.

MAXIM ALZOBA: And another question about the 10% of threshold which is allowed, and the operating window for -- yeah, the maintenance window. And for EPP for example, sometimes registries have windows of silence between different phases of registration and eats more than 2.4 hours.

FRANCISCO ARIAS: Yeah, so we're not currently monitoring EPP and I think what you are saying will happen only before general availability, right?

MAXIM ALZOBA: Between limited periods and things like that.

FRANCISCO ARIAS: Right. So we haven't started that, but the general idea at the moment is that we will only start monitoring EPP once you are in general availability to avoid these complications of dealing with previous periods.

MAXIM ALZOBA: Question about limiting window to 31 days. What was the reason? Because usually we want to track performance of our team and our systems over a year. Because usually TPIs are based on an annual basis or quarterly basis, and making 12 requests trying to catch the windows of the beginning and the end… what's the reason? Because usually we just track what happened over the year and we see a list of incidents.

So is it possible to extend it? Because from a matter of sending us 1000 characters or 3000 characters, it's not a big difference from a computer point of view. I'd say you will save a few bytes, kilobytes. I'm not sure it's the thing you need to do.

FRANCISCO ARIAS: You're right. It's about the resources. Not so much on the response, although that's also a concern, but remember this is a

query to a big database. And to give you an idea, the amount of data it's been generating, I think last time we checked it was on the 20-something terabytes of data generated per year, and it may be more now. So we're trying to limit the queries to something that is more manageable. Now, this is the history of the incidents you may have had in your services. If you are interested in getting an export of that, maybe we can give that to you offline through another method.

MARK ANDERSON: Thanks, Francisco. This is Mark Anderson from Verisign. I'm kind of jumping the line because I have a follow-up question on that. So I understand that you have 31 days of historical data through the API, but that you keep all the historical data in there, and it could be used for -- if we requested it offline, you could provide it in some kind of offline mechanism. Do I have that right?

FRANCISCO ARIAS: Yeah, I think we can do that. I would prefer that, extend this method, at least that's my initial calculation.

MARK ANDERSON: Okay, but at least for now, it's limited to 31 days? Okay. And so, you don't have any concrete plans right now to sort of show historical data, histograms, the like?

FRANCISCO ARIAS: Oh, this is historical data. This is Francisco Arias from ICANN org. This is historical data. It's just the window will be 31 days, but we intend to keep all the historic data. So once history is accumulated, you will be able to say, "Give me the 31st days of last year." And you will be able to see that. And then the next 31 days and so on. You could actually be moving from a 31-day window and obtain the data that way.

MARK ANDERSON: Okay, I got you. Thank you.

UNKNOWN SPEAKER: [Inaudible], DNS Africa. Morning, Francisco. Always good to be at one of your sessions this early in the morning, [inaudible] lack of sleep. On your new measurements -- thank you for that, [inaudible] increasing that. That's very, very nice. I was wondering on the granularity to go with that. So will you be able to give me information, for example, if I'm to query that the service is down over on IPv4, over on IPv6.

Can you give me a trace root from the probe, for example, to go with that because in there I want to make sure that I can pick up these things before you send me to some sort of threshold that you create an incident. I want to know exactly that information. I'm not looking for a lot of granularity from the measurement statistics. Can you give me that information?

FRANCISCO ARIAS: This is Francisco Arias. I think it would be available, if not explicitly, you can get to that because if you see here, there is a target IP address that is the main target for measurement and from the probe nodes leads you will see the IP address of the probe node that is querying. So you can see what's the issue between what and what IP address. I don't know, I think that gets to what you were asking. Thank you.

KAL FEHER: Kal Feher, Neustar. I think I might have a follow-up to that. I've been asking for a couple of years now, since this was run. Can we have baseline data? Baseline data. So you can only get this probe information after an incident. But I don't know whether one of your probes is almost triggered, I only know when they are triggered. I want to know when, for example, if latency is a particular level. I'd like to know what that level is during any

normal day. So the baseline monitoring data, not just when it's in breach of the threshold.

FRANCISCO ARIAS: This is Francisco. So you're asking to see the response time, for example, which is what we capture even if it's outside of the -- if it's not failing, if it's not causing an issue.

KAL FEHER: Correct. Yes, that's exactly it.

FRANCISCO ARIAS: Okay. Let me look into that.

KAL FEHER: Okay. And the other question I had was if we ask for downtime, would I be correct in assuming that that will not -- sorry, for maintenance -- that that would not be added to the downtime value? You would still derive the downtime value based on observation rather than our submission?

FRANCISCO ARIAS: This is Francisco. You're asking if we are going to count the downtime during a maintenance window?

KAL FEHER: No. To be specific, if I ask for a maintenance window of, say, four hours and the service is actually offline for, let's say, three and a half or some fraction of that, the downtime value that you present will be incremented by three and a half, not by four?

FRANCISCO ARIAS: Right. The downtime is counted from what the system is seeing. It's independent of the maintenance window. It doesn't look into that at all. The maintenance windows are only to suspend triggering alerts.

KAL FEHER: Okay, thank you.

CRYSTAL ONDO: Thanks, Francisco. Crystal Ondo, Donuts. If you have multiple TLDs, how are we supposed to tell you about maintenances? Do we have to do it 238 times, for example? Or can we do it once for all the TLDs in the system?

FRANCISCO ARIAS: This is Francisco. Unfortunately, at the moment, it's per TLD. We don't have the contact model, and we're relying on the GDD

portal contact model. It's per TLD. So we don't have that functionality. Now, the potential silver lining here is that we're talking about an API that you are, I suppose, going to automate this. I mean, there will be some [inaudible] for, but it should not be that complicated to automate it. But we can certainly look into individuals for a better solution for registries like you that have multiple TLDs.

CRYSTAL ONDO: And in the meantime, can we still just send emails? I mean, right now I send an email that says we're going to be down from these times for this many TLDs.

FRANCISCO ARIAS: Right. So you continue to send the emails, but we are not intending to stop triggering alerts in those issues. Yes.

SAIF AHMED AL-MASHHADI: Hi, this is Saif, dot IQ manager. Regarding the access to your system, what are the requirements from our side? Like zone access, DNS access, with bind, if there is any requirement to access our zone so you could monitor it.

FRANCISCO ARIAS: This is Francisco Arias. Are you a ccTLD?

SAIF AHMED AL-MASHHADI: Yes.

FRANCISCO ARIAS: No, we don't need to get access to your zone. If you like to provide it to us, we would be more than happy to take it, but it's not a requirement. Yes, Peter. We're interested in [inaudible] for study, analysis, and only for that purpose, to be clear. We have talked about this with the ccTLDs before. But it's not a requirement at all for getting access to the system. You just request access and we will give you access. DNS services, of course, is a public service so we're not expecting to be needed to be whitelisted to monitor it.

SAIF AHMED AL-MASHHADI: Okay, thank you.

MAXIM ALZOBA: Maxim Alzoba for the record. It's a follow up to the question about the requests of historical data. Have you checked which causes more load? Twelve requests in a row, or one? My experience in database says that one is probably going to be

safer. You can check and see the difference. And since we're going to request it, like a few hundred of us, it might affect the load to the database and front end. Just a suggestion. It's up to you to follow or not.

And about incidents during the maintenance window, as I understood in the past, maintenance window incidents were marked as false positive. Or I was under the impression that it was this way. And currently as I understand, all you do is suppress sending us messages. Am I right?

FRANCISCO ARIAS: This is Francisco. No, we don't usually use the false positive functionality. We avoid it unless it is a false positive, and they are not common. If a registry were to ask us now to suppress the alerts, we have a way to do it. We have functionality in the system to say suppress alerts to the registry or to compliance or to my team. There are certain levels that we can define.

So we have a specific functionality to suppress the alerts now. What this API is doing, that is exposing that you so that you can directly set that without having to request to us because we don't have, currently, an automated way to do that.

MAXIM ALZOBA: Yes. For clarification, for example, we have 40 minutes of service downtime allowed per week, just four hours. I'd say 24 hours for something like RDDS. And we request maintenance windows of three hours. Does that mean that during this seven days rolling week those three hours are going to be marked as breached? Not breached, but effectively wasted SLA time.

FRANCISCO ARIAS: I'm sorry. I do not follow.

MAXIM ALZOBA: For this service, during the seven days, we're allowed to have up to 24 hours of failed minutes of tests, effectively.

FRANCISCO ARIAS: In a given month, and it's not 24, I think it's 16. It's 864 minutes.

MAXIM ALZOBA: Okay. And if we take, for example, one hour for the maintenance window, does it mean that this one hour is deducted from the allowed time we have?

FRANCISCO ARIAS: Oh, I see. This is Francisco. No. The maintenance windows, again, are only to prevent triggering alerts. The count of downtime is that. When we see downtime, we deduct that time as you said. But a maintenance window does not deduct anything from the allowed downtime of services that have that.

MAXIM ALZOBA: And another question about nodes. Do you see that names of the nodes have different IP addresses all the time? And seeing that, it's going to be quite useful for us to see the name of the node, or maybe even we don't need it. We need an IP address because to start an investigation on our ISP side, all we need is an IP address, autonomous system number. And if you give us the name of the node, we have to get a check of what's behind this name. And given that, you give us historical record with the incidents, and your list of nodes to the IP address conversion is the current. It'll give us, actually, no access to what was in the past.

For example, you change IP address of Washington DC node in one month and then a month passes, and you change it, and we check the request, and when we pull the IP addresses from that file, we actually have all the data because it changed. So please provide it via a core or something to avoid a situation where you unintentionally provide us with something which is not true.

FRANCISCO ARIAS: Thank you, Maxim. This is Francisco. I think I got what you are saying. I think the issue you are raising could be solved by adding another field in this measurement to say this is the source IP address of the query. I'll look into that. I'm not sure we will be able to do it because we are very close to our release time, but in the next release we'll look into that. Thank you.

KAL FEHER: Kal Feher, Neustar. I've got a related question to that. In the original pilot, I remember that there were quite a few probes that were offline. When I say quite a few, I think it was like 3 or 4 of the 50. Will there be a way of querying the long-term status or the current status? Is there any way of knowing how many of the probes are currently offline? And if there's any way of maybe tracking that back to even our own systems, if that's relevant. Or if the offline status is purely unique to the probe itself.

FRANCISCO ARIAS: This is Francisco Arias. There is no method considered at this point to offer that. I'm trying to understand what is the use case that you are thinking. It is true, that we have a set of probe nodes, as you said; not all of them are active at any given time

because we do maintenance, or they fail or something. What is the use case that you have in mind?

KAL FEHER: Well, it would be useful to understand if, for example, you're looking at an access problem that's not universal. So if only a few nodes are in the area and a probe that's also offline might be considered to be a similar region or a similar connectivity path, then it might be, especially when you're talking baseline data in the future, if we can receive that, then it might be useful to take into account when assessing the overall performance over time of the system.

Because we presume that 50 probes are hitting our systems, and if it's not 50 probes then we would have to effectively adapt our expectations of the performance based on that, of what we've seen, that is, based on that. I may not have explained that correctly, sorry. I can think of the use case, I just can't express it.

FRANCISCO ARIAS: Okay. If you don't mind, maybe you and I can talk offline.

KAL FEHER: Sure.

FRANCISCO ARIAS: Okay, thanks. Do we have any more questions? No.

MAXIM ALZOBA: Maxim Alzoba for the record. As I understand these particular implementations, it gives us not just read-only things, but also we can send you notifications of maintenance windows and things like that. And since it might influence our performance under the contact, is it possible to make a simple send box with like three incidents? The TLD, which is non-existent, so we can mingle with it and understand if our part of the interface is working correctly without sending you false maintenance window items and things like that.

Maybe limited, then you see that -- something like OT environment for the system. So a registry can check that, yes, the engineers did all the tests, they tried everything, and they successfully created maintenance windows, tried to delete it at the wrong date. So you see that your system is working. I think it's not very wise to try it on a live system.

FRANCISCO ARIAS: Thank you, Maxim. This is Francisco Arias. I'll look into that option. I think we provided something like that for the RRI, so

maybe we'll be able to provide something like this for this system.

Now, I just wanted to clarify something. I don't think that most of the methods are read only, the only ones that are write are the maintenance windows. You can create a maintenance window. But I don't think it affects your contractual requirements because it only suspends the triggering of the first alert for RDDS.

MAXIM ALZOBA: Actually, we have to notify our registrars at the same time. What's the reason to poke registrars for testing purposes? When we have a maintenance window, we notify you and registrars at the same time to avoid different issues later. Because when something starts, like an investigation of our activities, we provide the proof that yes, we notified all the parties to our contract.

If you see this as a formal method of interaction, please think about a checkbox. You have a limited window of tests, like two days. You have to run through those tests. And something like registries do. So you don't have to keep lots of instances. For example, we will stand in a queue for the ability to check it on send box, if you can.

FRANCISCO ARIAS: Like I said, we will look into providing an [inaudible] environment. Thank you.

No more questions? Last call for questions or comments in the room. Kal is requesting to present this to the Tech Ops group. I'm sorry, I have to confess, I don't know what that is.

MARK ANDERSON: Thank you. Mark Anderson. So it's a technical group within the Registry Stakeholder Group. That's an excellent suggestion. Maybe you and I can talk offline and figure out a time when we can invite you to present at the Tech Ops group. Yeah, thank you. Great suggestion.

FRANCISCO ARIAS: Okay. Thank you, Mark and Kal. Sure, I'll be happy to talk with you. Last call for questions, comments. No? Okay. Thank you very much. Yes, Maxim.

Maxim is asking about the documentation, and I missed including that in the presentation. As soon as I have the committed date for release, we intend to publish the final spec, just to make sure that we are publishing the final version and give you some time to implement before we make it into

production. But yes, we intend to share that and will send communication to the gTLDs, and in general, to everyone through the gTLD tech mailing list. Thank you.

[END OF TRANSCRIPTION]