MONTREAL – How it Works: Understanding RDAP
Tuesday, November 5, 2019 – 15:15 to 16:45 EDT
ICANN66 | Montréal, Canada

STEVE CONTE: Okay, we're going to get started. I thank you all for either staying or coming to the session. The session is going to be on RDAP and the RDAP protocol. We do have some table space available still. The How it Works session is meant to be a dialogue and meant to have that kind of communication going on. So, don't feel like you need to be audience, please come in and be a participant. And you know, ask questions as they come up and we'll try to get to them.

I'm going to be looking away from you, so I apologize. So if a question is back here, raise your hand. Gustavo will see that and smack me and tell me to come and bring you the microphone. So we'll do that. Paul, is that a question? No, I'm kidding.

So this session is on the RDAP protocol and we have Gustavo Lozano. See I practice and I still got, Lozano. Gustavo Lozano from our Global Domains Division; Technical Services. He and his team have worked on the RDAP protocol with respect to ICANN and the domain stuff. And I'm going to pass it over to Gustavo to take it from there.

GUSTAVO LOZANO: Hi, hello. So yes, as Steve mentioned, this presentation is about RDAP and the protocol that is a hot topic right now. So I'm going to start with the presentation. Feel free to raise your hand and I will try to answer your question as soon as possible.

So this is the agenda for today. So first I will go to an introduction, then we'll start the section about what is the difference between WHOIS and RDAP. Then we are going to go deep into the RDAP protocol. I'm going to explain what is the RDAP profile and the next steps and what are some of the tools that you can use on this transition.

So it's an introduction, right? Right now, on ICANN, there are two lines of work related to registration data services or what you may call WHOIS. One line of work is how to replace WHOIS, how to implement RDAP and how to deploy this new protocol on the gTLD space. That's one line of work. It's quite technical. And the second line of work is in relation to the policies that are required to define who is going to have access to the data, how do you transfer the data, how do you display the data and the reason to get a unified access model and all those things.

This presentation is about the first topic, is not about policy. If you are here to -- I mean if you are in this place to about policy, that's not going to be the case. I have so many slides about an overview of how the policy is going on but that's it. I mean, we are not going to talk about that.

So I'm pretty sure that you're familiar with this diagram that we have on the screen. It's how registration works on the DNS space right now.

You have a Registrant; the registrant wants to get a domain name, the registrant contacts that registrar, provides some information and then the registrar uses EPP and the domain name is registered in the registry, right? That's pretty simple. I think that everybody's familiar with that.

But now what is the Registration Data Directory Services? So that's the green circle that we have here. And this is the service that you use to get the information about the domain name, right? And that information could be technical information like the expiration date, the creation date of the domain name or it could be related to the contact information of the registrant that registers the domain name.

Right now the protocol that is used it's called WHOIS. And the idea is at some point to replace this protocol with this new protocol, okay? Pretty simple. And ICANN is not within the path of that registration or access to the information. ICANN is outside of this picture developing the policies on how this interaction works. So I think that's clear to -- hopefully it's clear to everybody here.

So a little bit about history. So back in 2010, the work began to replace the WHOIS protocol. The Regional Internet Registries, the registries for IP addresses and autonomous system numbers, they have been using RDAP for years. And some registrars are still also offering services on RDAP. So based on a decision, it started and now we have a protocol that is defined with some specifications called RIRs. And since the 26th of August of this year, gTLD registries and registrars, they are required to implement an RDAP service.

There is still some work going on. For example, right now there is something called the gTLD RDAP profile that defines how the protocol is supposed to work and interact on the gTLD space. But we are working with the contracted parties to have this profile as a requirement. So that's something that we're working on.

We are also working on defining the service level agreement for this new service. If you go to the recent agreement for the new TLDs, you will notice -- and also for some legacy TLDs -- you will notice that there is a service level agreement and some service level requirements for the services like DNS and WHOIS. Well, we also need to define those same service level requirements, but for RDAP.

We also need to define the reporting requirements. Some of you may be familiar that if you go to the ICANN site, you can get reports from the gTLDs regarding the number of DNS queries, WHOIS queries that they receive. So we want to also have some kind of reporting requirements for RDAP. And eventually, at some point, WHOIS is going to be retired. We have a page, a microsite on ICANN and you can access this link to get more information about this protocol.

So what are the features of this new protocol? And also when you look at the features, you will understand why we're doing all of this. On RDAP we have a standardized query and response mechanism. It's based on HTTP, which is the protocol that is used to basically run the web. So if you access a website, you will use the HTTP protocol. We have secure access to the data, so basically we have encryption. Obviously, you need to use HTTPS. There is extensibility. This means that the protocol can be extended to support new elements if that's a necessity.

And it enables to have differentiated access. In other words, you can have limited access to anonymous users or you can have full access for authenticated users. Whatever that definition of authenticated users is at the end.

There is a bootstrap mechanism to find the authoritative server. This is really important because right now for WHOIS, it's not that easy to find let's say the server that I need to query in order to get information. But in the case of RDAP, there is a mechanism that has been standardized to find that authoritative server.

We have a standardized redirection/reference mechanism. So what this means? This means that if you want to get information from the registry and the registrar regarding a domain name, there is a way to go from the registry to the registrar.

As I mentioned before, it uses HTTP, which is a well-known web protocol. It supports internationalization and there is some support for searches; I'm not going too deep during the presentation into what search means. Questions, until this point? Nope. Okay.

So now I'm going to talk about policy. As I mentioned before, I have a few slides about policy. This session is not about policy. So in May 2018, the ICANN Board adopted a Temporary Specification for gTLD registration data. This was an interim solution or measure to help with compliance regarding GDPR. This policy triggered a policy development process to confirm if it's a consensus policy or not within 12 months. And the first phase of this work is what you call the EPDP 1. And implementation is going on.

We have the EPDP Phase 2, which started in March of this year. And the idea of this PDP is to consider if there is a way to define a standardized access mechanism for non-public registration data. The work is going on and there is the idea that there is going to be a report sometime at the end of the year.

And if you want to learn more about policy and this EPDP Phase 1 and Phase 2, there are several sessions. Some already happened on Monday. There are others that are going to that are scheduled for Wednesday and Thursday. And obviously, you can go to the agenda to see when those sessions are going to take place.

And I want to mention before I get into the protocol that work that was done by the ICANN TSG. So the ICANN TSG was the Technical Study Group that was formed by ICANN and the objective of this group was to develop a technical model that could be used to provide guidance to Phase 2 of the EPDP regarding how a Unified Access Model service may look like. So that work was finalized in April, 2019 and you can read about this model on the link that is on the presentation. This is all that I have about policy. So I'm going to go into what the protocol is and how it works.

So this is a comparison between the WHOIS protocol and the RDAP protocol. So in the WHOIS, you have a plain text response. In the case of RDAP is also text-based but it's machine parsable. And this is really important. What machine parsable means? It means that a computer can take the output from this protocol and it's able to identify which elements are, or the contents of the elements on the response.

So it means that if you have a response from RDAP, the machine is able to understand that www.icann.com, that's the domain name. So that machine parsable is -- that's what it means. And this is the most powerful feature of the protocol because once a computer is able to understand the response then you can have all kinds of applications or clients that can use this information.

So in the case of WHOIS, the format differs from server to server. And in the case of RDAP, that's not the case. There is a format that is already defined.

RDAP is flexible in terms of fields and functionality and the response from the server can be easily converted as you desire.

This is an example of a WHOIS output. I think you're pretty familiar with that output. This is the output for icann.com for that domain name. And this response was provided by the very same WHOIS server because [inaudible] registry. And this is the same content but in RDAP. So you can see that -- well, maybe for a human it's not that easy to read, but for a computer it's fairly easy to parse and to understand.

And this is a raw input, right? You can have this input and you can have something like this in which it's let's say easier to understand for a human being, but it's the same output. And obviously, the idea is not to show this to the end-user, right? The idea to have this output from RDAP and then you have a [inaudible] client, like the kind that we provided on lookup.icann.org that is able to query for the domain name using RDAP and then it will show a human-friendly page like the one that you have here.

So the RDAP protocol is defined. I mean, the actual protocol is defined on a set of documents that are called RFCs, which stands for Request for Comments. And these are the main documents that define that specification. As you may notice, these are the base documents. And then the work continues on the protocol and we have new specifications being defined as we speak for the protocol.

The protocol was defined on the IETF. What is the IETF? It's the Internet Engineering Task Force, it's a group of engineers that work together to create the standards. And those standards are used on the internet, or usually on the internet. There is a working group -- I mean within the IETF, there are working groups and each working group works on a specific let's say topic or on a specific protocol. And there is a working group called the REGEXT Working Group, and that working group continues to work on extending the RDAP protocol. Questions?

So on RDAP, there are two main concepts regarding how to get the information. The first is a lookup and a lookup is a query for a specific object. So if you want to get information for icann.org, that's a lookup and you will only get information for that specific domain name.

And we also have a search. So what is a search? A search is a query that as a result may provide several domain names or several objects. An example of a search is, 'tell me all the domain names that are registered by organization X or Y', or 'tell me all domain names in which the name server icann.org is the name server of the domain name'.

So you have lookups and you have search. The search functionality that is defined in the base protocol is quite limited, and a lot of work that is going on right now on the REGEXT working group is to define better search capabilities. There is work going on to define reverse search for example and so on. So lookups are clearly defined and search is a topic that continues to be in development right now.

So on RDAP, you have base objects of information. And those are domain names like icann.com. Nameservers like ns.icann.com. Entities; in the case of RDAP, all the contacts like the admin contact, technical contact, registrants, all of those are called entities. An organization is an entity.

Then you have the autonomous system numbers and IP networks. The last two are used by the Regional Internet Registries, by the RIRs. And the first three are also used by the Regional Internet Registries but these are the main objects that are using the gTLD space or in the domain space.

Domain name queries. This system it's probably the most interesting object for this community. So a domain query is used to identify a domain name and it's used to get the data for the domain and the domain name could be provided as an A-label or ASCII label or as a U-label. So for example, this is an A-label and this is a U-label, and this is an IDN in A-label format.

This is what is called the base URL. So once you know the base URL, let's say that the base URL for that com is rdap.nic.example, then you just add a '/domain' slash the domain name that you want to get information for, and you get information for that object. So it's pretty simple. Once you know this base URL, you just add '/domain', slash the object that you're querying for and you will get information for that particular domain.

So once you get a response and do you remember the response that they show you at the beginning of the presentation? That response that is machine parsable, that defines engagement.

So what is JSON? JSON stands for JavaScript Object Notation and it's a way to structure data or it's a language to structure data for transmission of data between computers. JSON is widely used on the internet right now and there are other languages to structure data like XML or YAML. But at some point, it was decided that JSON was the most appropriate language to define RDAP.

And on JSON you have sets of name and value pairs. So you have for example, in this case, this will be the key that you normally see on WHOIS and this will be the value for that key. And you have types of values like number, strings, Boolean, array, object.

This is an example of JSON. And you can see that -- I mean, probably it's -- I mean, someone can understand the contents because at the end of the language, JSON tried to be easy to read by a human being. So for example, in this case, you can see that this is information about a book, right?

So once you query for the meaning, you get a response. And what you see in the response in that JSON format is the following, right? You have a handle and the handle it's a unique identifier for that domain name. Then you will have the name of the domain name that's in this member called LDH name. So basically it's called LDH because it's Letter Digit Hyphen. So here you will have the domain name, let's say in ASCII format.

Then you have the Unicode name. So if the name is an IDN, you will see the U-label on this member. You have variants in case that, this is IDN and then you have variants for that IDN. And then you will see the variants on this member. You have the nameservers, the entities, in this case, the entities are the contacts like the admin contact, [inaudible] contact, registrant; even the registrar is an entity that is associated with that domain name. And you can have information regarding network there.

We have information about DNSSEC. So if that domain name is the DNSSEC signed, then you will see all the information about the public key and that member.

So if I go back, remember that we have the domain name and we also have entities. And I mentioned that entities are the abstraction for the contacts that you normally call contacts on the WHOIS, like the admin contact, registrant and all of that. So in the case of an entity, we have again a handle which is an identifier. And all the information about the address, telephone number and all of that, that are encapsulated within an object called a jcard.

So jcard -- and sorry if this gets too technical. So if you use outlook or any kind of email application or agendas and all of that, when you send the information of a person or a contact to someone else, that information probably is using vcard as the format to transmit the information. So it was decided that we should reuse that same format for RDAP. So right now on RDAP if you get a response, you will see that there is a jcard object for the contact information.

You have the roles, you have entities for that entity. So what it means, you can have, for example, for an organization and that's an entity, you can have entities below to describe for example the legal contact for that organization for example. And you have other members that are used for the IP community, the IP address and community.

This is an example of a jcard. It's probably kind of simple. You have a version of the jcard or the vcard and then you have the full name. You have your organization, you have the telephone and then you have something called the address. This is a member, and within the affress member, you have the actual address for that person. The address is 'adr'. So it means that it must have a certain order to be parsable. And the first element is the post office. Then you have the extended address, then you have the street, the city, the state, the postal code and the country name.

And regarding the country name, I don't know if you remember, in WHOIS, you don't get the country name for the country, you get the two-letter code. So the RDAP protocol was extended fairly recently to support two-letter codes for the contact instead of the country name.

So right now if you get the RDAP response in the gTLD space, this is going to be empty or null and then you will have a new element describing the two-letter code for that country.

STEVE CONTTE: Can I have both?

GUSTAVO LOZANO: I mean you can have both, but the profile recommends that once we have the support for two-letter codes, which is the case now, you should use that one.

Name servers. So as I mentioned before, you have domain names, you have entities. Another object that is used on RDAP, it's the nameserver and the nameserver is, as the name implies, is to provide information about the nameserver of the domain name.

These are some examples of name servers. All of these are valid and this is the A-labeled representation. And this is using also the A-label for an IDN. And you have your labels in the last one.

And these are the members of that object. Again, you have a handle, which is a unique identifier. You have the name in LDH format, you have the name in Unicode, you have the IP addresses in case that the nameserver is suddenly recalled or you have IP information for that name server. And you can also have entities.

And finally, we have these objects that are used for the Regional Internet Registries. Probably these are not that interesting for this community. But you can also use RDAP to get information about an IP address or a range of IP addresses.

So this means that if you have been attacked, for example from a specific IP address, you can get information about who is the party responsible for that IP address on the internet. And this is the way that you create the query. You specify '/IP' and then you specify the IP address or you can specify a range of IP addresses.

These are some examples of IP addresses and range. And these are the members for that object. You have again the handle, which is the unique identifier and you have a start and address, in case that you're looking for a range. And then you have the ipversion of the IP address and the country and more information about that IP address or range.

And again, if you want to get information about an autonomous system number, you can also use RDAP for that. An Autonomous

System Number is an identifier that is used to establish BGP connections between ISPs on the internet. I am not going to enter into that but it's basically the protocol that is used between ISPs to connect on the internet. And again, you specify the '/autnum' and then you specify the ASN and you get information for that ASN. You can specify 16 bit and 32 bit ASNs and you will get the information.

Another feature that is really important regarding RDAP is that we have standardized error codes. In the case of WHOIS, it was open to every TLD to define how to display those error codes. Error codes like a domain name don't exist or whatever. So if you go to WHOIS, you will see different variations of that error code. And in the case of RDAP, we have standardized error codes. So that means that a computer understands that a domain name doesn't exist, right? It doesn't need to read the contents. You just need to look at the error code to find information.

Internationalization. This is also a really important feature of RDAP. Right now in WHOIS, there is no way to support internationalization. It's an ASCII protocol basically. In the case of RDAP it, it uses UTF-8 and because it uses UTF-8, you can have most of the world's writing systems or yeah writing systems, you can encode them in UTF-8. So that means that RDAP supports internationalization out of the box. That means that if your name uses non-ASCII characters, you will see those non-ASCII characters on the response without any issues or your address or even the name.

Bootstrapping. This is also a really important feature for, I mean on RDAP. And if you have read an [inaudible] agreement, you will notice that it says that they should provide the WHOIS service on WHOIS.nic.TLD. I mean, when the agreement was defined, that was a way to create some kind of mechanism to identify the server in the case of WHOIS. But it doesn't work as well because in the case of legacy TLDs may be 'nic.tld' is already registered.

But in the case of RDAP, we don't have that kind of officious. There is, if you go to the IANA website, you will find on this link what we call the bootstrap. And that bootstrap is a list of all the TLDs and what is the RDAP server or the base URL that you need to use to get information for that TLD.

Extensibility, as I mentioned before when RDAP was defined, the idea is that you can extend the protocol fairly easy. So in the case of RDAP, we have what we call a registry for extensions. And that registry is on this link that you can see on your screen. And it's a list of all the extensions that have been defined from registries and registrars to encode new types of information.

So if you have an RDAP client, you can go to that website, to that link that is there and you can see how to decode new types of information. Like for example, if for a TLD you have union special contact because that contact is defining someone that lives within a city or whatever, then that new type of information is going to be defined there.

These are some of the proposed RDAP extensions that the IETF is working on as we speak. So for example, we have or we want to have support for RDAP Reverse Search Capabilities. So what is a reverse search? This is, give me all the domain names that are registered by this person, for example, that's a search capability.

We want to have support for partial responses. Again, if there is some reason why you need to provide let's say not all the elements for the domain name, but you only want to provide some partial information, then the idea is to support that on RDAP.

There is also support for sorting and paging if you have like a lot of results and there is also this really important draft that is trying to define how to support federated authentication on RDAP.

What is this? Right now the way authentication works on RDAP, if there were a policy to define that when a user username and password you can get full data in RDAP, you will need to get a username and password for every TLD on the internet. So you will need like thousands of credentials, right? Once this draft becomes a standard, you may have one user name and password. You are going to be able to use that one on any TLD on the internet or gTLD, or whatever the policy defines.

Questions at this point? Yes, please.

BILL SWEETMAN: I can see many benefits to the RDAP protocol. But a question I have is it also one of the benefits which might be a negative is that it appears to me to make it easier for bad actors to potentially scrape or even be more effective at scraping WHOIS. And I'm just wondering to what degree the protocol anticipates how to block or inhibit bad actors from scraping WHOIS?

GUSTAVO LOZANO: There is nothing on the protocol in that regard. I mean, the only thing that is in the protocol it's not in the protocol itself but on HTTP. You can [inaudible] limit the number of queries that you receive. That's basically what is there. That's something for policy to define. I mean, policies should be created regarding let's say the amount of access or who has access in order to prevent that kind of bad behavior. But yeah, I mean, on the protocol there is nothing there to prevent any kind of scraping let's say. And as you mentioned, maybe it's easier because you have only one format instead of different formats.

PAUL WILSON: Paul Willson from APNIC. I heard something a little while back that the use of vcard was being reconsidered. As far as you know, is vcard, jcard now a fixed and done deal or is it still under some doubt?

GUSTAVO LOZANO: There are some proposals to have a different way to encode the contact information, but I mean, even if that draft becomes an RFC, it will need to be mandated to be a reality on the service itself. But yeah, I mean there is some work going on because developers are not happy with jcard, I mean some developers.

PAUL WILSON: One of the things that is potentially important and hopefully will be implemented in the RIR world is the use of multi-lingual content, is populating some of our WHOIS data with multi-lingual content so that they can be both a Latin character and one or more local character versions of contact names, organization names, addresses and so forth. And that at least is something that vcard, jcard supports. And I hope it would be maintained in any alternative, particularly if it's sort of mandatorily replaces vcard or jcard.

GUSTAVO LOZANO: Yeah. In the case of the gTLD space, there is work going on regarding transliteration and translation and the idea is, if you do translation or transliteration or if you have the data in different, let's say languages that you need to provide those language texts on the vcard so you can have information about the specific language for scripting. So yeah, there is some work on the gTLD space.

PAUL WILSON: It's not so much a case of translation or transliteration because there are literally multiple addresses. There'll be literally an English language address and a Chinese character address and you might translate them from one to the other but they actually are not exactly translatable. They are just two versions and both of them need to be stored. And that's the existence of multiple versions in different character sets, is what's important to preserve.

GUSTAVO LOZANO: Any other question? Nope? Oops, sorry. What is the gTLD RDAP profile? So the way the protocol is defined in the specification, it's like a Swiss knife. I mean you have all these kinds of objects, all these kinds of functionality but there is no definition of what functionality should be supported or how things should work so you can have an interoperable service.

So this is what the gTLD RDAP profile is trying to do for the gTLD space. The idea of the gTLD RDAP profile is to define a way to provide a service so that you have interoperability between different parties.

So as I was mentioning, RDAP is a real flexible protocol. It allows implementers to choose from different features and the idea of the profile is to define which features are on, which are off, those kinds of things. And the idea in the case of the gTLD space is to map that current policy and contracted language to requirements for an RDAP service. That's the intention of this profile.

There was a discussion group from gTLD registries and registrars and they developed this RDAP profile. The RDAP profile for the gTLD space is basically two documents. One is the RDAP Technical Implementation Guide and the other one is the RDAP Response

Profile. And those two documents define how the service should be implemented and operated in the case of gTLD registries and registrars.

The issue is that for now, compliance with the profile is recommended and the community is working on the contracted language and obviously an amendment to the contracts of the contracted parties so that the profile is required.

So what is defining this profile, and this is really important because this is basically the requirements for registries and registrars in order to have interoperability between the parties. So it defines what fields should be included in the response. It also defines that if you want to add new fields, you don't need approval from ICANN. You can extend the fields without any kind of approval.

It defines which objects are supported like domain names, nameservers, those kinds of things. It requires to support the service both over IPv4 and IPv6. It defines some HTTP headers that need to be provided in order to have Javascript clients. Those are the course headers for those of you that are technically inclined. So the idea is that you provide those course headers.

It defines that you need to support DNSSEC. It defines that you should reject queries if you're mixing A-labels and U-labels. It says that registrars need to provide the base URL for the RDAP service. It includes terms of service, that you should support help queries. So if you go to the base URL and put a '/help', you will get help regarding that specific server or service. Truncated responses, use of country codes instead of country names. URIs to facilitate communication with a contact, the URL for the AWIP, the compliance URL.

It defines the requirements for, to register RDAP extensions and DNSSEC elements. Registrars should only respond for names that they are the sponsoring registrar using registrar identifier for contact and so on and so on. So it's, so those are two huge documents and the idea is to have interoperability. They are not required by contract right now, but I have been testing several services and it appears that that community is implementing the profile even if it's not a contractual requirement.

References from Registries and Registrars. As I was mentioning, this is one of the things that is required by the profile. And this is, in the case of .com and .net' and the thin registries, you get only the thin registration data for the domain name. So the rest of the data is with the registrar. So this is a mechanism to go from the registry to the registrar to get the full data for the domain name.

The protocol supports HTTPS. So you have all the benefits of HTTPS, you have encryption, you can even have authentication using TLS, I mean HTTPS if that were the case.

Redaction requirements. So as you're maybe familiar now, some of the information or most of the information regarding the registration contact information is redacted. In the case of WHOIS, you have an element like the name or the street and then you have a value of redacted for privacy. In the case of RDAP, the way it works, you don't have the elements. So basically you won't see the fields let's say that is redacted and you only get a remark saying this object has been redacted due to authorization. That's a difference between how WHOIS and RDAP works.

And differentiated access. So the protocol supports differentiated access. That means that if you are able to authenticate yourself and the server authorizes you to get more information, you can get more information. This is already supported. But obviously, policies need to be defined, to define who is going to get access and so on. RDAP protocols support differentiated access out of the box.

So what are the next steps? ICANN is working as I mentioned before with gTLD registries and registrars to define service-level requirements like round trip time, availability, downtime for the RDAP service in the gTLD space. We are also working on reporting requirements. This is going to be really important to get a sense of how many queries are being done on RDAP, how many queries over WHOIS.

We want to have some contractual language so we can enforce the RDAP profile. And obviously we are working with the contracted parties on retiring WHOIS. At some point, WHOIS is going to be deprecated on the internet, well from the gTLD space.

Tools. There are a lot of tools, servers and clients, open source. This is a list of clients that you can use to query RDAP. So you have the ICANN's RDAP web client, CentralNIC, there are a lot of clients on the internet. I think that there are some even apps for phones right now to use RDAP.

And now the idea is to do a simple demonstration on how the web client that we developed works. It's pretty simple. This is ICANN's web client. So you can enter a domain name here and what it does, it goes to the bootstrap. And remember that I mentioned that there is a bootstrap mechanism.

Using the bootstrap, it finds that you are querying for a domain name under .com, then it gets the base URL for .com, it gets the information, let's say the technical information, and then the client goes to the registry to get information about that registrant and the technical and admin contact. Obviously, because of the Temporary Specification, most of the information is not there because it has been redacted. But you can see other information like the information about the registrar, DNSSEC information, authoritative servers.

So what you see on the screen, we are basically merging the response from the registry and the registrar into one big response and using a human-friendly interface to provide you information.

Down below, you can see the actual response from the registry. So this is the RDAP response from the RDAP server of Verisign obviously because this is a .com. And if you go below, you can see the response from that registrar. And this is the JSON for that response. And that's it. This is is RDAP. This is how you get the response and JSON from the servers.

There are also command line tools and apps, there are different clients that you can use. This is the end of the presentation. Any other questions?

BILL SWEETMAN: Bill Sweetman from Name Ninja and I realized I didn't identify myself when I asked my first question. So a clarification and a question. Did you say that for fields that are redacted that the output doesn't even display sort of the category?

GUSTAVO LOZANO: So within the RDAP response, you have objects, right? So for example, the entity is one of those objects and then you have and so on, so on. So if you want to redact the object, what do you do on RDAP is you basically don't put the object, right? So you don't have anything indicating that the object is there or not. But you get a remarks member saying, "Hey, this object that you are seeing here, that object has been redacted because of privacy concerns."

In the case of WHOIS because it was line by line and you have fields on every line, then it was pretty simple to define that line. I mean the field is there but obviously, the value is redacted. I mean, I don't think there is an issue with -- that is just the way the protocol works because you have objects within objects, right? It's only that.

UNKNOWN SPEAKER: Under GDPR, obviously a lot of stuff is under privacy. And so we send emails like cease and desist letters to the privacy address, and if you delete an object then how do you do that if the email that's under privacy isn't actually there?

GUSTAVO LOZANO: Well right now, you will get the -- obviously you send these letters to the registrar to get information or --

UNKNOWN SPEAKER: Yeah, but the registrars have particular email addresses for their privacy. You can't just send it randomly to the registrar.

GUSTAVO LOZANO: I think so far there is a way to provide that, let's say privacy address within the registrar or the privacy proxy provider in which you can send the email to get more information. I mean there is a way to put a remarks member for that. Sorry if I'm confusing things with -- if I'm creating confusion by saying that the object it's not there; is just a technical term of how it works. But you will get the information of that privacy address to send the email to get more information.

UNKNOWN SPEAKER: Also is there any protocol for doing bulk downloads within the system?

GUSTAVO LOZANO: No, there is no support for bulk downloads. I mean if by bulk download you mean that you send a request and you get thousands of objects, that's not in the protocol right now. I mean there is support for search but I don't think that anyone is supporting search right now. I haven't seen search support, but search is different for bulk download, right?

UNKNOWN SPEAKER: They are similar. You've got 500 domain names, you want to know the WHOIS for all of them and who to send the emails to, how do you do it?

GUSTAVO LOZANO: There is no way to do that.

PAUL WILSON: There actually is a bulk response format. There wasn't originally but APNIC is part of our historical WHOIS implementation, we've got something we call 'WhoWas', which is an ability to sort of issue a WHOIS query and have it return what the history of responses to that query were over time; and we implemented a bulk response which actually is just simply a concatenation of a bunch of records with the right sort of header information. And I'm pretty sure it's standardized.

GUSTAVO LOZANO: Isn't that in the base protocol?

PAUL WILSON: No, it's not. But it wasn't originally, it had to be added at some point.

GUSTAVO LOZANO: Okay, well, yeah. I mean obviously the protocol can be extended and nothing precludes some kind of bulk mechanism to be implemented. But right now in the based protocol, there is no support for bulk download. If by bulk download you mean like sending a request, like give me your whole database and getting that database, that's not there.

DAN YORK: Dan York, thank you very much for this presentation. I've been hearing about RDAP for ages and just have never actually sat and looked through it. So it's super helpful to understand the pieces. But I do want to build on what Bill said here earlier. I do find it fascinating because I just played the command line client right here and I can see, I mean I like the fact that it's a very restful kind of format and you can just go and use the URLs.

But for somebody to gather data out of this, I can just pull the thing from IANA of all of the RDAP servers and then I can just basically have a little script that can go and query those to '/domain', slash whatever and start to pull down all sorts of -- so I mean it's just interesting how it's been done. I do think that there will need to be defensive measures designed on the RDAP server-side because it would be easy for attackers to go and just trying to kind of walk up the tree by brute force to go and see what domains are there.

GUSTAVO LOZANO: But you can also do that on WHOIS. I mean, it's human nature, right?

DAN YORK: But this just makes it kind of a little bit easier because it's all standardized in a standard format versus how the WHOIS servers and the different ways and there's a central repository of RDAP servers in ways that were not necessarily there in the past. So, interesting.

PAUL WILSON: Security by obscurity has been fairly well debunked, isn't it?

UNKNOWN SPEAKER: So to follow onto the previous two questions. I mean, if this is the big benefit, this is machine-readable and machine-parsable, are there plans to actually make this available as an open API spec so anyone can write to it? So then you could eliminate manual WHOIS searches and just set up an API query, just sort of update something and --

GUSTAVO LOZANO: So the service is an API. So the way it's defined is an API already. And maybe you are talking about libraries; there are a lot of libraries right now for different languages to play with RDAP but the definition of the protocol itself is an API. It's a restful API basically.

BILL SWEETMAN: Question. Bill Sweetman from Name Ninja, again. I can't believe I'm asking so many questions about something so arcane but oddly I find this fascinating. I don't know if this is contained within the jcard aspect of the protocol. But I'm wondering if there's any I guess you would call it error protection contemplated in the protocol? So for example, phone numbers; does the protocol in any way check the structural integrity of the phone numbers? Are people allowed to put a dash or not put a dash?

I'm just wondering if any of that is contemplated because I can see how it accurately presents data but it doesn't mean the data is accurate or in the right format. So area codes within a phone number, for example, because within the current WHOIS protocol people, unfortunately, put things incorrectly in all the time. And I'm wondering if our RDAP helps make that phone numbers, for example, more accurate.

GUSTAVO LOZANO: I mean if by accurate you mean that they are formatted in E.164, I think that's the ITU standard for phone numbers. I don't remember if that beaker specification defines that if the number is not in A.64 -- I mean, in whatever format they define, it's going to get an error. I can look on that and answer that question about the syntax or the phone number.

Now, there is nothing in the protocol saying that if the country is USA, the international area code should be one. No, that's not there obviously. But regarding syntax, I can take a look at the specification and I can make a proper response. I don't remember right now.

STEVE CONTE: Any other questions for Gustavo? Gustavo, thank you very much for a wonderful presentation. Please. For those of you who have been with me for most of the day, for the How it works stuff, hang in there. We got one more at five o'clock in this room for the root server tutorial. It'll be a presentation from RSSAC and we'll have root server operators here to answer any questions about the root server system and service if you have any. So please join us back here in 44 minutes, and thank you very much.

[END OF TRANSCRIPTION]