MARRAKECH – Tech Day
Monday, June 24, 2019 – 13:30 to 18:30 WET
ICANN65 | Marrakech, Morocco

EBERHARD LISSE: Good afternoon, everybody settle down please and sit down. [AUDIO BREAK]

Welcome everybody, my Eberhard Lisse, I'm the Managing Director of the ccTLD Manager of .NA, I am Chair of the Technical Working Group, which organizes the 39 Tech Day. To go through the agenda as usually, I will give a call out to the people I don't' know, which I don't know or who I don't know whether they are here while I go through THE agenda.

First, we talk a little bit about RDAP and Mario Loffredo is speaking, is he here? Can you come already sit here because we can start right away. Then we will talk a little bit about Linking Blockchain to DNS and Brantly Millegan is there, you can also mossy up to the front if you want to. Then Jacques Latour will give us an SSAC update on Internet of Things, I have seen him, I don't know whether in the room, there he is.

Patrick Jones will speak about the DNSSEC training that ICANN and the Regional Organization have provided, he doesn't need to raise his hand, I see him there. Then we have Jay Paudyal is here, Jay are you here, raise your hand please? We don't see him but we'll see whether he's maybe busy with something else and will come later through the time.

Otherwise, the next would be John Levine, I've seen John during the day. He will talk about email and the DNS and I will hope he will explain to me what [inaudible] and all these funny things that prevent me from sending mail to Google are.

Then Jaap Akkerhuis has pointed us to the website internet.NL, which is a nice little tool to check your domains things, domains and email addresses and so on. I thought we'll let him give a presentation. Everything in blue is a clickable link, so if you click on internet.NL you'll obviously get and have a look at that website. Then we'll have a presentation from Egypt about IDNs and generalized acceptation. Is the presenter there? There we go.

Then we have Jiankang from .CM, do I see him? Not yet. He has presented in the past about this topic and he sent his paper in so I'm sure he will be there on time. Then we have a problem in that the speaker from .MR who told us and I quoted that we can count on him for a presentation, sent us -- did not send us a presentation and he did not inform us that he's not coming, which I find borderline rude, even though I'm a guest in this region.

Fortunately, Tim April had asked me a few days ago whether we would like to listen to his presentation, the one that he made in the morning? If everyone who is here in the afternoon has heard it already, he will go through it much faster but I'm very grateful that we have somebody to fill in the space. As you can see originally, I didn't put a break, it's always a good problem to have when you need to go into the tea and coffee breaks. Finally, then Ondrej Filip will give some remarks.

Last time, Warren fortunately gave me his notes from his remarks and so I produced a report which will happen this time again. This will also be posted; the abstract will go to the ccNSO for their meeting report and the report in total will also be presented. We will try to put the links to the presenters in there as well, so that if somebody wants to refer to it, that would be helpful.

MARIO LOFFREDO: Hello everybody. I'm Mario Loffredo, I'm a researcher at Institute of Informatics and Telematics of the National Research Council of Italy. The Institute is located in [inaudible] the .IT registry. In this presentation made in collaboration with my colleague, Maurizio Martinelli, I talk about our RDAP implementation experience at .IT. I talk about four RDAP applications, namely validator, crawler, server and the client, which we have developed but they are still in progress. Finally, I say a few words about what we're going to do in the next future.

The first application I want to talk about is the RDAP validator; it verifies the response compliance with both RDAP and jCard specification, is based on the json schema, the last version. Any RDAP application developed at .IT has been written in Java and its implementation has been quite difficult due to the many requests for comments and standards it takes in consideration. There are a lot of RFCs involved in RDAP and as many in jCard and a lot of RFCs and standards referenced by both RDAP and jCard.

The second application is a crawler, it's based on the RDAP validator. It checks the responses coming from the service included in the IANA Bootstrap Service Registries. It performs a validation according to three steps; the parson according to the json schema, the validation against the standard profile, some rules written in RFC7482 and 7483. And the last validation against the gTLD profile according to what is written in two documents, the RDAP technical implementation guide and the RDAP response profile. These parts of the application are still in progress.

So far, the RDAP crawler has discovered the following issues. I mention only some of them, grouped by category. For example, the issues about jCard, we found some other servers returning the jCard fn -- which didn't return the fn format name element, which is required. For example, they returned the TL element, including URI type but returning any the URI value. We found some issues about the standard profile. For example, some other servers returning good values unregistered or returning and error code, an error response as a string instead of a number. Other issues about the gTLD profile, for example the IANA registry, which is not registered or the lack of the domain registrar contract. General issues like server didn't return an answer or didn't return any valid content.

As registry, the many application related to RDAP is the RDAP server. The RDAP server has been quite challenging in mapping between the .IT data model and the RDAP data model, especially with regards to the map with the RDAP entity objective. It provides search queries only to authenticated users. It returns different content according to the user provides, it supports boot strapping. Currently it is based on the .IT public test environment registration data. The data are not real, are fake but we plan to move this RDAP server on live environment before the end of the year. Now, it is available at the URL, you can see in the slide.

It implements several extensions. I have mentioned only the extension whose development .IT team is directly involved. Some of these extensions have been described in many ITF drafts and there has been submitted to the ITF. The ITF rejects the working group for standardization and the first three extension are now at the stage level of proposal standards.

Let see in more detail some of these extensions. The first is about the efficient management of large contents, which can be returned by the other servers as response of a search query. As you know, in RDAP you can submit a search but the server can truncate the response according the server limits. You can have the drawbacks that you cannot see the real amount of results if there are truncated. So we thought to an extension enabling the users to obtain all the relevant results. We define in this extension and this draft, three new parameters. Count, which allows the user to obtain the total number of results, which can be by itself relevant information. Sort, which allows the user to sort the results. And cursor, which is an opaque string encoding the pagination information. No matter if you are implementing the typical offset limit pagination or the key set pagination.

New properties have been added to the standard response. Sorting metadata and paging metadata, containing respectively the information about the current and the available sort criteria and the total number of results and the paging information. So, two new text have been added to the RDAP conformance array.

This is an example of this sorting metadata information included in a response. You can see that the server outlines the current sort and provides a ready-made reference to a different view of the same results according to different sort criteria. This is an example of the page metadata, providing information about the total count results found and the link to the next page of the results. We transform a sustainable search for the server to a sequence of sustainable search for the server.

This session is about the possibility to request the server for a partial response. As you may know, now in RDAP it is not possible to request the server for obtaining a subset of the full response. You can receive only the full response. The basic idea behind this proposal is to enable the user to ask for a subset or the response but not declaring specifically the data fields you want but declaring a server to define a set of data fields. We have introduced a new parameter called field set, it's a string that defines set data fields. In the draft we have recommended the RDAP providers to implement three field sets. Id containing only the key field, brief, identifying a set of fields conveying a basic knowledge of each object and full field set containing all information the server can provide according to the user profile.

The standard response we have a new property, called the subset in metadata, including information about both current available field set and the new RDAP conformance tech.

This is an example of sub setting metadata, providing the user with information about the current field set and an alternative view of the current result set in a different view.

The subsequent extension is about the diverse search, which give the user the possibility to obtain the list of domains related to an entity, which is for validity. For example, for the end all, for a name, for the email, for address. In the draft we defined four new search part segments. These are examples of the query that the user can submit to the server to find for example the list of domains whose technical contact is an entity having an email as specified. Since this draft have submitted it has raised some concern about the risks related to the privacy. The authors have clearly defined the draft that these capabilities are not open to everyone. It must be allowed only to authorized users. We're just submitting the queries under some lawful basis. For example, a classic scenario is registrar searching for their own domains or predators in the size of an official authority or performing a specific task in the public interest. We think that the privacy considerations are now past and we are focusing our selves on the technical issues.

Another extension is about the possibility to submit complex queries. As you know in RDAP now the query are very simple. You can not submit a search query combining [inaudible] joined in end, for these reasons very big result searches. We hope this extension as mean to provide restricted result search. To achieve a more precision in the query submitted. We have new parameters, query, which allows the user to submit the complex search, it must be used in place the define RDAP search part, segment part. The filter which allows the user to filter the results according to the values of those properties that are not used as search part segments. These are examples of the queries that can be submitted by using this capability. You can have the maximum freedom to submit the query you want to receive to relevant data.

Another important extension, we are now working is the possibility to have a different contact representation other than jCard. If you know jCard, you know that jCard is quite inefficient because it cannot be deserialized by using the standard methods offered by the json libraries. You have to customize your deserialization method to obtain an internal representation of the jCard. Now we are collaborating with Robert Stepanek from Fast Mail to provide a new contact representation alternative to jCard, provided that jCard is the default representation method. We defined a new extension, a new query parameter that is called JS Contact. If you submit a query JS Contact equal to default or the default query, you'll receive the jCard; otherwise, if you specify jscontact=true, you will receive the representation according to the JS Contact representation.

I skipped the domain name extension due to lack of time because I want to focus your attention on the last extension which is the specification extension. You know that now there is a great consensus about the fact that IP service should provide clients with a machine process specification capabilities. They should describe more formally the records they accept, they response they provide and the authentication methods. There are a lot of API specification languages, unfortunately neither of them is standard. Anyway, they describe the same object. The languages are very similar and there are automatic tools to make a conversion between specification languages into another. We have defined a new end point called specification.

The client can obtain through this end point a form, a specification of the server capabilities. We think that this extension could bring benefits, both to the server and the client. The service side, the server can provide a machine processable specification of the records, the responses and the supported authentication method. It can announce to clients any change about its capabilities and make it suddenly available. At the client side, the client can configure itself according to any service specification and user access level. The user must doesn't need to know all the capabilities of the server. Clients can enable only -- the user to submit only valid request and display and valid the responses more efficiently. This is response of the specification and the point. You server can provide the list of a specification, the specification according to different form.

Connected to this last extension, there is our version of the idea of the RDAP client. Now, the RDAP server can be pretty different, both in requests and the responses but can provide the machine processable description they don't feature. The current RDAP clients, which providers user with capabilities because we are all based on the standard specification. They are based on RFC7482. As a consequence, the users might waste time submitting invalid requests that cannot be satisfied because they are not permitted at all or because they are not allowed according to the user access level. The users and the clients must know the feature of what the server they interact with. If the server makes a change about its feature, such a change is not immediately recognized by the client and this results in additional effort by the client implementors.

Our idea is to implement a client able to configure itself according to the server specification. It will be based obviously on the server specification extension. It could be able to generate the user interface automatically from the specification it receives from the server. It can take advantages from the availability of libraries performing the conversion automatically between a specification language. This is scheme of the progressive steps of the client. The user selected the target server, the specification is requested to the server, the specification and the point is searched. If now specification is available the standard specification is loaded. It behaves in the same way as the current RDAP clients. Otherwise, a specification is available, the client search for the open API specification language that we have elected as our internal format. If this specification is not available and no open API, it can be easily transformed in open API. At the end, the client interface is generated by the client, it's what the user interface library. At the end, the user can submit only the query it is allowed to submit to the server it's interacting with. The development of this client is still in progress.

A few words about the future activities. We want to move forward with the current IETF draft. We are evaluating the submission of new IETF drafts. We want to give our contribution to the jCard fix replacement issue. We want to complete the validation against the RDAP gTLD profile. Obviously, we want to complete the client. Finally, we want to migrate our server on the live environment. As I told you before, we plan to do that before the end of this year. Thank you for your attention.

EBERHARD LISSE: Thank you very much. It has, though it didn't need to, convinced me that RDAP is very complicated, in fitting with what ICANN usually does. We've got a few minutes for questions. Please feel free to come to the microphone. Identify yourself for the remote audience please. We can't hear you; can we have the microphone turned up, please?

RICK WILHELM: Rick Wilhelm, Verisign. Good presentation, thank you. You mentioned at the beginning you were crawling and bumped into issues. Just curious, were you able to send contact the server operators to let them know about their issues?

MARIO LOFFREDO: We are starting -- the best way to provide this capability. I don't know if it will provide by a webpage. For example, you submit your response, a example of the response of your server, so you can have back the response of the validator or -- you can submit for example, the URL of a request and the crawler is able to parse the response and give you an example [inaudible] for example. We are evaluating the best way to provide this capability.

RICK WILHELM: Because on the ICANN gTLD side, as you know because you're deeply involved but for those in the ccNSO that aren't as close to it, the RDAP production date for gTLD's is in August and to some people you're probably hitting in production are probably not yet don't have the production code up. If you need help getting contact, you can ping me because we do the RDAP pilot working group, maybe we can help you get in touch with some of the server operators, so we can work on helping to sort out their questions. Then also, this is probably more for the sake of the audience as opposed as from Mario's perspective, you had talked about the reverse search and you had considered the privacy considerations worked out. I'll just offer for the assembled group, that that's a very localized consideration, not necessarily a gTLD consideration because RDAP and privacy, RDAP pilot working group is taking its lead from the EPDP implementation review team on that and that of course gets into the unified access model. The RDAP pilot working group is following in their lead but certainly appreciating the work you're doing there. Thank you.

MARIO LOFFREDO: As you know, this draft is still under discussion in the ITF working group, so we are open to every suggestion, preferably about the technical consideration because we are convinced about the fact that every RDAP server is subject some local laws about the provisioning of sensible data. When we vote about this capability, we had clear in our mind that this capability would not have been accessible to everyone, only to some specified roles.

EBERHARD LISSE: Any other questions? Alright, thank you very much. That was quite educational. Next is Brantly Millegan, who will give us a technical non commercial presentation about Linking Blockchain to DNS.

BRANTLY MILLEGAN: Hello everybody, this is my first time actually attending an ICANN event, so I'd be great to meet some of you guys. My name is Brantly Millegan, I work for a nonprofit called True Names Limited, it's based out of Singapore and we just focused on developing what's called the Ethereum Name Service, which is an open source project, it's public domain, all the code is public, anybody can contribute, people can folk it if that want. Our website is ENS.domains. It's called the Ethereum Name Service because it's built on the Ethereum Blockchain and not because it only serves the Ethereum Ecosystem, although that is our focus right now. Our lead developer is Nick Johnson. Our initial patron was the Ethereum Foundation, I believe a Swiss nonprofit.

He's an outline of what I'll be talking about. I'll first just give a quick look at the blockchain naming space and how ENS works, just to give some context but then I'll be primarily focusing on our plans for DNS ENS domain integration. I'd actually like some feedback from who are here, if you could help us with some of this.

Really quick on the blockchain naming space, it goes back all the way 2011 with the launch of Name Coin, which was the first fork of the bitcoin code and they had .BIT. They had the idea that you could do this. I'm not sure that Name Coin is really been super successful in that but they're still around. The Ethereum Name Service launched in early 2017 with the native TLD .ETH and we'll be adding other ones as well soon, which I'll be talking about. There's EONS, EOS Name Service, focusing on EOS ecosystem although I think they may be running into some collision problems. Unstoppable Domains in a new thing, it hasn't actually launched on Jane, it's running on Zilliqa, they have .ZILL, it's VC backed. There's a bunch of other ones, one for the Rootstock Network for Ethereum Classic, they're all pretty small.

Right now, ENS is the one with the most success by quite a bit. We have 275,000 names registered. A dozen wallets supported. We have native integration at the offer browser. You can also get that in Chrome, Firefox Edge through this MetaMask plugin, which is primary way to people can turn their browser into an Ethereum enabled browser in general. That also connects them to ENS. We have a partnership with IPFS. We actually just announced that Encirca is the first ICANN accredited registrar that we're aware of, that also now offers .ETH addresses. ENS just in general I think benefits from the growing Ethereum Ecosystem, which is the largest smart contract blockchain platform by a long shot. Cloud Fair just announced an Ethereum bridge, Microsoft is very involved, Google has Ethereum linked to their big query. Samsung just launched with their new phones a cryptocurrency wallet that only supported Ethereum. HTC Exists is dedicated blockchain phone. There's a lot going on here.

Just really quick about how ENS works. The first thing to understand is that there are no servers. Sometimes people think they need access to our servers or our system. We don't run any servers for ENS. Our website is obviously on a server but other than that, which you don't have to use to access ENS, there is no servers. There's no permission needed to use the system. If everyone ever wanted to do anything with it, you can just do it and that's because it runs entirely as a set of smart contracts on the Ethereum Blockchain, the whole ENS thing. This is how it works, it's very simple.

There are two parts to the system. There's the ENS registry which has .ETH TLD and it has the second level domains. Let's say if I want to resolve example.ETH, I first send a message to the ENS registry and get back the address of the resolver and we separated these two steps for a very important reason I'll explain here in a second. Then you go and you find the resolver, you find the ENS records and you bring back your record. All of this by the way, when you're doing a look up, does not require a transaction, it's free. It does require transactions and transaction fees to set up ENS records and register your name initially but all resolution is just a look up. This is ENS, you can look up any information you want.

I'll talk a little bit about ENS records. We separated out the two steps so that people could actually create any kind of records they want. You can create your own ENS record set and put any information there for any use case. Now, we've created what we call Public Resolver, which is kind of the resolver that we created, that anybody can use and it just has three fields, an address field for an Ethereum address, that's what you use to send tokens. A content field for IPFS or swarm hashes, those are both decentralized file storage systems. IPFS is Inner Planetary File System, they're trying to compete with the server system. That is also not natively supported by Opera. If you type in a .ETH address that has an IPFS hash, you will resolve an IPFS website. No normal server involved at all. Then we have a place for text records. We plan on adding an optional WHOIS, although we don't have that right now, we're going to use the text records. We're also working on adding native supports for DNS record types. We may also offer support for resolving cryptocurrencies, their addresses but, we don't have to be the ones to this, anybody can do this.

The main thing here. The goal of our DNS EN integration is this. We want to make it so that people can make ENS records for DNS domains that they already own, through the normal DNS registration system. Here's what this means. You have example.com and you have a DNS record, we'd like it so that example.com can also have and ENS record that lives on the Ethereum Blockchain. This is could be a mirror of your DNS record, maybe as a failsafe, a backup, that's one possibility, or they could just be other types of records that DNS is not normally used for. This is not so that you can have example.ETH, so you can example.com and ENS record, it's a common misconception when I talk about this. For example, very basic use case. You could create an ENS record for example.com that has an Ethereum address and so you could be using any of the main Ethereum wallets and you could send tokens to example.com or you could resolve an IPFS website, you could do anything you want. You could of course put an IP address in an ENS record and resolve that, use that rather than using DNS for that lookup, you could do anything you want.

It's a two-step process of how this works. Step one, you need to prove ownership of your DNS domain to the ENS contract on Ethereum. We don't anything manually, this is all done automatically with smart contracts. Step two of course is that you need to create and you manage your ENS record on Ethereum. Step one really is where the meat is, I'm not to talk about this here for a few minutes. We've decided to use DNS SEC for proving domain ownership. I'll assume that many people here have an idea of how DNS SEC works, it works with a recursive cryptographic proof, so if you want to prove ownership or let's say data in a text record let's say on a third level domain, you sign it with your private key for that but then that needs to be signed by the second level domain, the top level domain, all the way back to the DNS root to prove ownership. We've actually created a DNS SEC oracle on Ethereum with a DNS root public key. This is a smart contract that has the DNS root public key built into it and you can submit any DNS SEC proof to it and it prove that now to the Ethereum Ecosystem. We're actually really excited about this, this could have a wide range of use cases, many of which we probably haven't thought of it. We're using this right now only for proving ownership so that you can set up an ENS record for a DNS domain but I'm sure there could be many use cases here. It also saves proofs already submitted and that's useful because it can make future proofs cheaper to verify. To do a proof you do have submit a transaction which means you have to pay for it, it could be very cheap, the more complicated it is, the more expensive it is. We're talking like maybe rather than 10 cents, it's maybe a couple dollars or something but if save proofs already done, if you're doing something that involves that information, it doesn't have to reprove or verify that part of it. For example, if you've already proven ownership of .com, you don't have to go all the back to the ENS root, it can stop at the .com stage of the proof because that proof is already on the Ethereum Blockchain.

This is how we do it with second level domains. You have to create an underscore ENS example.TLD subdomain. You need to have at least one text record field that has lower case A equals and then insert the Ethereum address that you want to own, own and control the ENS record. If you don't already know on Ethereum, on your account in some sense, is a public private key pair, you control the private key obviously and keep it private, the public key is public, people can grant ownership of things to your public key and then you control it with your private key. Here you're putting an Ethereum address which is actually a hash of a public key but basically like your public key, you put that there and if the system proves ownership, it will automatically, we're not doing this, automatically grant ownership of that domains ENS record to that address. You submit the proof to the DNS oracle. Like I said, yes this does require having an Ethereum Wallet and some eth or ether to pay gas. Right now, you can actually do this with some domains, we've running tests, I'll be talking about them here in a second. It's kind of hard to do because we don't have an UI built for it, so you kind of have to just know what you're doing to do this directly but we are working on UI because we'd like to make this very simple for people to do. If the proof succeeds, it grants ownership to that Ethereum address as I explained.

If you claim an ENS record for a DNS domain, we are not charging any -- there's no registration or annual fees at all. This is unlike .ETH names, which we've created, we do have an annual fee, for most it's like five dollars a year, you pay an eth and you can pay ahead but we're not doing that for any ENS records for DNS domains and part of our thinking is that since you've already paid fees, you own the domain, like we don't need to charge, you don't need to pay a fee to create a record for a name you already own, and that's kind of part of our bigger philosophy, we're nonprofit, we're not trying to, you know, there's no investors, we really like to see ourselves as part of the global namespace, I'll have a few words about that at the end. We'd like this to all work in a fair, reasonable way. The reason we even have fees to begin with is actually just to try to prevent name swapping, that's it. And so five dollars we thought was about the smallest amount that would still do that, but not so much that it would prevent legitimate use.

So then you have to create an ENS Record. This is actually fairly simple, we have a great UI for this, manager.ens.domains. You have to have an Ethereum-enabled browser, so that could be Opera or Chrome, Firefox, Edge, with a Metamask plug-in which has your Ethereum account built into that. It just takes a couple transactions. So, we've actually tested this, this has been running now for a while. We contacted the .xyz people and said can we use your namespace as a test? And I think they were kind of like, sure, whatever, you know, we don't really care. So we've been doing that. So you can claim second level .xyx domains on Ethereum right now, in fact we have services using this.

There's a wallet called Argent, based out of Europe, and they have Argent.xyz as their website, then they've also claimed it on ENS and they give sub-domains as Ethereum account names to all their wallet users, and then they also use that sub-domain on DNS to give them a user page; I think it's a very clever cross use of the system. So this has worked flawlessly over the last year or so. We're very excited this has worked well. We also like to have not just second and lower integration, we also would like to have top level domain integration, as well. Second level domain and lower integration could be launched for all properly DNSSEC-enabled domains right now, and there are about 1200 of them. There are a lot of country codes that don't, but hopefully that will change over time. But we'd like to get the top level domain situation figured out first, and let me explain why.

Just as a second level domain owner can claim that, or a third level domain, once you own that, of course you are able to control everything lower than that, because DNS is hierarchical, so is ENS. So if a top level domain owner claims their top level domain, that gives them control of their entire namespace on ENS. Now, why would you want to do this? Well, we've actually tested this, as well. So, .luxe, they were very excited to be the first top level domain to do this. They claimed their .luxe name on ENS so they controlled the whole namespace. It's right on their website if you go to nic.luxe, it explains what you can do there. Why do you do this? One, you can make the process easier for customers to claim their ENS names, since you control it. So they don't necessarily have to have an ENS account at all, you can control that on the backend for them if you'd like. You can also just kind of manage their namespace as it exists on ENS if there's something that you feel you need to do to curate it in some way. So there are these two approaches, either one, you don't claim your top level domain, you just let second level domain owners do whatever they want, or you do claim it, and you can use that to kind of curate your namespace. Like I said, we really would like to respect the ownership that people already have in the namespace.

We have one problem, though, with our top level domain integration. Top level domain owners of course need to be able to prove ownership just like second level domain owners, but we're not exactly sure how to do this on a wide scale. We can't use _ens.TLD for the text record, of course, because my understanding is you're not allowed to use second level domains that begin with an underscore. And then we thought that all top level domains had nic.TLD reserved, but apparently they don't, apparently only new top level domains do, older ones don't, like .com, which is very important, and then many country codes don't. We weren't aware of this, so this is how .luxe works, that they prove ownership through their nic. But Verisign was very kind and they read our documentation and I believe it was Burt Kaliski sent us a letter and said this is not going to work, and here's why. So, thank you, Verisign, if you're here, appreciate that. But you also can't make a text record in the top level domain itself, that's not allowed. So, here are some possible solutions.

One is that we just manually approve ownership of top level domains, I mean, there are about 1200 with DNSSEC, you know, add in the rest, if they eventually add that, it's just a lot of work, it's going to be slow, we'd like to have this automated, that would be our preference.

Another idea we have, is that we could require a signed business but unpublished text record on the TLD itself. I've talked to some people about is that possible to do with the way people manage their private keys? The way those systems work, I'm not really sure, if that was possible, that would be really easy. If that's too hard, we may need to come up with something else.

And then if people have other ideas, we're all ears, we really would like to resolve this in the next couple months and launch this on a wide scale. So we're trying to figure that out.

Just a few quick words here before I finish up about ENS and our relationship with ICANN. We like to be respectful of the existing system. We don't want to pollute the namespace, so we did create our .eth name without ICANN approval, this was back when it was just a side project of a few people two years ago, we weren't sure how successful it would be, it has been fairly successful. So we'd like some grace for that, but otherwise we don't plan on making any other TLDs outside of the normal process. Also, we want to learn from ICANN's hard wand bureaucratic processes over the last 20 years. We would actually even like to use them, we'd like to see if there are ways that ICANN can bring its oversight into the blockchain naming space in ENS, we're open to that, we would like that. We'd like ICANN to have oversight in a future in this space and we're just here for dialogue and learning, that's what we're here for. Thank you very much. Here are some useful links. Thank you.

EBERHARD LISSE: Thank you very much, that was very interesting. I just checked, there is no .ETH in the root. [AUDIO BREAK] I can put any text in [AUDIO BREAK] my copyright for example is a text record. I'm not sure whether I can start with an underscore, but I can put text record in first level domain, top level domain names. So, if maybe you choose something else. If you tell us you must put in a certain -- what some other do is they tell us you must put this number in and then when they read it they know we have got control. I don't know. So maybe you must look at this and as far as nic.tld is concerned, ccTLDs can do whatever they want; ccTLDs have got a bilateral relationship, if any, with ICANN, and no form of best practice or control from ICANN about technology. Many of us adhere voluntarily by what the GNSO does so that we don't duplicate services and we make it easier for registrars, but that's the way life is.

BRANTLY MILLEGAN: Yeah, that's true. Our concern was we couldn't necessarily assume that it would be available. They may have already given it up.

EBERHARD LISSE: Okay, I've got three questions, and not a single more. The one in the back was the first one.

VITTORIO BERTOLA: Okay, thank you, I'm Vittorio Bertola from Open-Xchange. I didn't know this was coming up here today, so I actually replied this morning to your colleague on the DNSO. The only caveat I wanted to give you is that you seem to be approaching this problem from the technical side, but it's mostly a policy problem, and especially I would be very wary of that. It's unclear whether you just want to do a one way mapping from the DNS into the GNS which might be easier, or whether you also want to modify stuff in the ENS and push stuff back into DNS, including maybe doing registrations or transfers, but the more you go into that, the more problems you're creating at the policy level and I would advise you, you have written contracts with whatever TLD you want to bring into this. The only technicality I wanted to say that you said we want to verify this once and then it's done forever, but that is not how the DNS works, you have to [inaudible] so please respect them.

BRANTLY MILLEGAN: Yes, great question. So, the direction is only one way, so for example if I claim example.com on the ENS record and then I release it in the ENS system and somebody else gets it, they're able to claim that record and do with it whatever they want. It doesn't go the other way. So the control, everything is still managed in the DNS system, it just allows you to create ENS records.

EBERHARD LISSE: Two short questions, two short answers, please.

UNKNOWN SPEAKER: Yeah, hi, I'm [inaudible] from India. I have a question, what I have learned is if I have a .com I have ENS, as well. What if I abuse someone's trademark and how do you handle those issues in future as ICANN handles it via UDRP, Uniform Domain Name Dispute Resolution Policy.

BRANTLY MILLEGAN: For DNS domains that are claiming ENS records, because that is still managed in the DNS system, all that still applies. Because whoever owns the DNS domain is able to claim that so that is still managed by ICANN. If you're asking how we do that with .eth, specifically, which is not in the DNS system, our plan is to have a client side blacklist system that people can use to solve that problem. The blockchain naming space does have some unique advantages and challenges when it comes to the trademark dispute, but our plan is to focus on the client side blacklist system.

UNKNOWN SPEAKER: But how it, would, okay.

BRANTLY MILLEGAN: We can talk afterwards.

EBERHARD LISSE: It's easy to communicate offline later or by email, but I would like to stay within the timeframe. Jacques will talk to us again about the internet of his house.

JACQUES LATOUR: Hello, my name is Jacques Latour. I'm with SSAC. And today we're going to talk about the Internet of Things and DNS. SSAC released The Internet of Things and DNS Paper, it's SSAC 105. And this was written by The Internet of Things working group within SSAC. So, a little bit about SSAC. We have 39 members. We generally write recommendation and advice paper to the Board, and this paper that we wrote is a little bit different than that, and that's why I'm here to talk about that. And then you can see what our expertise is and all the documentation that we've done in the past. So, the paper that we wrote, in the introduction part, we talk at IoT and DNS. We talk about opportunities for the DNS with the upcoming IoT wave. We talk about the risk that the IoT device can pose on the DNS, and that's partly a very important thing to look at, to see what the future brings us. And we talk about the challenges we have with the DNS and the IoT industry, and how we can today be ready for the future to have secure deployment of IoT device and IoT cloud services.

So, this talk is not necessarily about our secure home gateway, it's mostly about the work we've done at the SSAC working group. So, like I said, it's a different kind of report, it's about 20 pages, it's an assessment of of the current state. It's a forward looking view of how IoT should look. So, I'm just curious, how many of you have read the document? A few? All of it? A couple more. So hopefully after this you can access the document and go look at it. It's important that normally we make the document that we write our recommendations to the Board to look at something or do something. In this case it's more a document for the community for us to look at things and figure out how we can address the future, so hopefully we might have some working group or further work that comes out of this to fix or address some of the challenges and risk that we identify. A lot of what we're going to talk about in this paper, it's not new, but what we've done is put it in the context of IoT and what mass scaling of billions of devices on the internet infrastructure can cause or impact overall.

The definition that we have for an IoT, it's a thing that connects to the internet, it's got some network connectivity, it has some computing capabilities, but it's not a computer. An IoT device is an actuator, a center, a thing that cannot protect itself, and that's a definition that we use for this document. So, drones, IoT sensors, and a lot of these things are what we consider IoT device. Home gateways, routers, are not considered IoT in the scope of this work. So the difference with an IoT device is it interacts with people, it interacts with the physical world. It collects data, it sends data, it sends alert, so it needs constant connectivity to connect information from the physical world to the internet, and that's where the DNS kicks in, and we have to make sure that part is done right. A lot of this happens without user knowledge. The users don't know necessarily that an IoT devise is communicating, it's doing DNS query reads, it's a sensor that's connected to a water pipe, you can't really tell what the interaction is with the internet.

So the key thing here is we're going to have 20 to 30 billion IoT devices doing a lot of this in the background. So at the internet scale, that could have some impact on some part of the infrastructure, so we need to look at that. And then the other thing, some of these devices may have a long lifetime. So, however they were programmed at the time of production with whatever software libraries, we need to acknowledge that they could be operating 10-20 years without being updated and we need to understand the impact of that it could have on the future. So we want to prevent that, today we need to write good IoT devices that can update themselves, but that's not a guarantee, and whatever we have today could impact us in the future, so we need to take that into consideration.

So, when we look at IoT and DNS, so this end serves, they connect to cloud services or they connect to a service somewhere, and this IoT device will do DNS queries and depending on how they do those queries or how they have been programmed to resolve the name of the services, it could have multiple impact. So if you have 20 billion devices that send GPS location every second, and every time the software does a DNS resolution for that domain name before sending the data, that could be large scale DDOS to a DNS provider. So, that's kind of the worst case scenario, but there are lots of scenarios to look at.

So, we know for a fact that IoT devices use DNS and we want to make sure they do that properly. So, we look at the opportunities. So if we do it properly, if we use the DNS properly, we can do DNS query more efficiently. The risks, I talked about that a little bit, if you have 20 billion devices querying the same domain every second that could be a large risk to the DNS infrastructure and there are challenges that we have identified, and we need to address those. So, in the document, we have multiple case uses, and multiple diagrams of where IoT device, the little d1, d2, d3 boxes in the top part, and these devices interact with users in the real world and primarily these IoT devices connect to web services over the internet. So if the DNS, obviously you want to protect the IoT device and you want to make sure they connect to the proper cloud service. There is a bad actor in there and you want to make sure that they don't gain control of the DNS of the IoT device, that they don't proxy or they don't do any bad stuff from to impact the users of the IoT device. So we've looked at that, we got a good description of all the issues around that and what things we should look at in the future to make that connectivity secure. But if an IoT device is compromised, it means it's personal information that can be stolen without the user's knowledge.

And in the scenario below, there is a large scale IoT device that could use one gateway to access a cloud service and then if just one gateway is compromised, then it can impact a lot of IoT devices and their functionality. So, case scenario there, somebody takes over a gateway that controls the traffic lights, that could be disaster. So, the opportunities that we've looked at, there are a couple. So, we had to look at DoH and DoT because it’s the new thing. But one of the opportunities there is resolver verification, meaning an IoT device could be programmed to connect to a known resolver with DoT or DoH, and if it's a trusted resolver, then you can have a secure connection from an IoT to a trusted resolver, and you don’t need to worry too much about the man in the middle attacks with DNS.

So, that's one good opportunity. We've got to make sure they do DNSSEC to make sure you connect at the right place, but you don’t guarantee that, but at least you get the good information at the IP level. Relevant to another publication we just made, you want to make sure when you register domains for your IoT services that the hygiene around the registration is proper, that you use multifactor authentication and all of that to prevent a domain from being hijacked. And you want to know the opportunities to make it visible to the user and the environment on where IoT devices are connecting and what domains they are using, and all that, because the challenge is an IoT device is a sensor and you can't see anything, so having an application that shows where your stuff is connecting graphically could help users figure out, there are impacts or issues to look at. So there are opportunities to look at and opportunities to work on, and these are the things the community should take on and work and bring forward.

The risks, I think that was one of the challenges in writing the paper. We seem to focus a lot on all the different risks in the future that could impact the DNS from an IoT deployment point of view. We try to keep it simple. We try to focus more on opportunities and challenges that we need to address. The risks are well known and have been well documented in the past, but we highlighted a couple that we think were important to be brought forward. One is DNS-unfriendly programming. So if a public library to do DNS stuff is used by a million or a hundred million IoT devices and by default somebody downloads a bad library, that can have significant impact on the DNS. So we want to make sure that the libraries that are out there that are common for IoT developers, that somebody actually vets all of that and makes sure that it's operating properly, and it meets the security by design criteria. So, there are a couple examples in the document, we talk about the TuneIn application on I think it's i-phone, and that caused some mini attack for cloud service providers. So there are a couple examples like that.

DDoS, well we all know about that, we've talked about it many times, so IoT botnets based attacks, that's hopefully not too common, so we need better practice and better security controls to prevent those.

And the last one we talked about are DDoS amplification, so that's an old one, but it's still happening, especially when there is a lot of open resolvers on the internet that can amplify DNS attack, so we haven't fixed that problem. So, that's not an IoT problem, but having billions of IoTs leveraging amplification attacks with millions of open resolvers that can create a good recipe for disaster, so we should focus on that stuff.

And then challenges that we looked at, and that's where we tried to focus more of our attention. So it's building secure software libraries for IoT devices, so software libraries that do DNSSEC validation or can support DoH or DoT by using trusted resolver connection. I think that's one good approach to leverage this new technology to esnure that an IoT device can do a query to a trusted resolver and trust the response back from the resolver when it says like, don’t connect there, it's not good, you can actually trust it at that point. So, I'm not sure if you want every IoT device to do full blown chain of trust DNSSEC resolution, but if you use a trusted resolver, that's a good option there. Having more control for the user is another thing that we looked at, so we need to work on getting the IoT operating system and the CPUs to leverage up to date DNSSEC and DNS libraries and we've got to make sure we keep those up to date in the future, and there are a couple examples there.

Training, so we had IoT security training, we had DNS training for IoT project managers, for IoT engineers to write proper software. That's something we need to figure, how or what the program is around that. But certainly when you look at the risk and you have billions of devices, you want to make sure that whatever the default software that we make available to the IoT industry is as secure as possible, that it can be updated, so keeping it up to date, making it updatable, upgradeable, is an important aspect. So the days of limiting a DNS query to 512 bytes, we can't have that in IoT device in the future. We need the most recent information available for developers.

I think we also need to understand to make the industry overall understand how the domain registration works and the security around that. I think we can do a lot of work on that, making it more aware for IoT product managers and engineers and us, including us, DNS people, understanding what we can do to help IoT people.

So more challenges that we looked at, across DNS operators to share information on IoT botnets, so it's understanding if there is a DDoS attack happening what can DNS operator do to understand where the attack is, look at the DNS fingerprint, so that was a recommendation. It's more inside DNS operator trying to understand what a profile of an IoT botnet, what the

DNS fingerprint might look like, and mitigation methods around that. So that's something we need to look at.

The next one is working with ISPs and understanding the profile of large scale IoT attacks and through that to signal back to the home gateways on what we need to do to mitigate these things. The other thing that we looked at in the report is using spin, that's from SIDN labs or the secure home gateway initiative from SIRA. So, SIRA and SIDN were working, it's a shameless plug, but we have to do it, to work on a home gateway that can detect and mitigate automatically against IoT device coming from IoT device inside the home. So we need more and more vendor initiatives like this and probably make that the standard base framework. But if we have billions of devices and they're all behind gateways, the gateways should be able to automatically mitigate and respond to IoT based DDoS attacks or any kind of attacks, for that matter.

So that's one area we need to look at. There's a lot of getting in there, the change is the biggest challenge and supporting all this new technology. So, one idea we had is to build some sort of a dashboard, so a system to measure the evolution of the IoT. So if we could start collecting key metrics on different aspects of IoT deployment, how many IoT devices support the different kind of operating systems, all the different metrics there. And then over time, if we start today, in the future it can help us to mitigate different kind of DDoS attacks or whatever security issues that can come out from these IoT devices. So, the document is out, the link is down there.

Like I said, it's not a standard SSAC paper, it's more a forward looking view on the emerging issues with IoT device and security. We want to make sure, I think it's important that we together make sure that the deployment of IoT device is more secure in the future. There are going to be a lot of these things and we don't want these things to attack us, we've got to make sure they're deployed securely. If you have any comments or issues, let us know. So that's basically the quick summary of the work. How many SSAC members here are on the IoT working group? So a couple are here, if you have questions, you can reach out. That's basically it. Any questions? Go read the paper. Thank you.

EBERHARD LISSE: Alright, thank you very much. Any questions? We still have time for a few questions. Alright, thank you Jacques, thank you SSAC. Okay, now we have Patrick Jones to talk a little bit about the DNS SSAC training ICANN has done in the regions.

PATRICK JONES: Thank you very much. I'm Patrick Jones from ICANN Global Stakeholder Engagements. Thank you for letting me give a brief talk about our regional technical training. ICANN has been doing this for many years. Some parts of the community may not have the awareness of the scope, the various types of technical training we do as the requests come to us from TLD operators or organizations, government, academic institutions, and others. Probably the last time that this topic was discussed as part of Tech Day or as part of the ccNSO track was by John Crane at the ICANN meeting in Beijing. So I thought it was time to bring this update back to this community and give you an overview of what we're doing.

So ICANN Org and the community regularly conduct awareness raising and trainings around the world on various subjects relating to the unique identifier system, DNS security, DNSSEC deployment. These are in the form of webinars, how it works sessions at ICANN meetings, technical workshops that we do such as the recent DNS Symposium in Bangkok or collaborations with DNSOARC and others, and we also do talks at global law enforcement trainings. And there are many other types of capability building that we do.

I'm going to start to talk about the DNSSEC deployment trainings that we're doing in the regions. Some of this is around

DNSSEC for regulators and decision makers. Sometimes they invite ISPs and businesses to those training events. We also do hands on training, working with the TLD operators and other organizations to help with DNSSEC deployment and we've done a train the trainer program, particularly in this region, and that has been actually quite successful, and I'll talk a bit about that coming up. And you think about earlier, one of the other speakers was talking about the number of TLDs that are signed, I checked, we're at approximately 1398 TLDs in the root that have signed with DNSSEC out of the 1530 that are in the root zone, about 50% of those ccTLDs have signed their zones and there is quite a bit of demand that ICANN receives for DNSSEC deployment expertise, and there are other regional TLD organizations and others like the Internet Society and the regional internet registries that are also doing various types of training, so we're not the only one.

When we're doing a DNSSEC training, we have partnered for many, many years with the Network Startup Resource Center at the University of Oregon. They have developed this virtual training platform using a very small Intel Nook box and when you are in a lab and you can connect a network to that device, it's very easy to set up a virtual training platform for the participants that are in the room, you can use that to generate test zones, you can test running open DNSSEC test monitoring and key rollover. And so we have been doing that, and they also do their own training using that mobile virtual training platform. They use it for a variety of other types of training, so ICANN is currently partnering with them on DNSSEC but there is other training that they do, as well.

So, here are some examples of recent trainings. During this fiscal year we partnered with regional organization in the Philippines, we have done this in a variety of places throughout FY19. Last year I went to Ankara, Turkey, as part of the Middle East DNS Forum. Some of you in the room were at that Train The Trainer session and in that Train The Trainer program we walk through, it's a multi-day course of how to deliver a DNSSEC training, and that includes everything from how to set up the lab using the virtual platform and going through the step by step approach of how we're delivering the material. So as a result of that training, we have been quite pleased to see many of those participant have gone back to their TLD organizations and started to sign their zones, also helping deliver their own training for their own local community, and that's quite positive to see.

It's often difficult to draw a straight line from the delivery of our training to a ccTLD or an operator to their decision to sign their own zones or turn on validation. But we have seen quite a bit of uptake within the last probably six to nine months in the number of ccTLDs that have signed their zones, quite pleased to see, for example, Kuwait and Algeria signing theirs, Bashar from the Kuwait ccTLD was one of the participants in our workshop in Ankara, so we are able to see that there is a direct benefit and while the operators, it's their own decision of whether they implement or not, when we have delivered the training, people are going out and using it.

So, here's some data on where we've gone during FY19 and these have covered a variety of types of subjects. Some of these have been covering DNS fundamentals and the DNS ecosystem, subjects around DNS abuse and misuse, and overview of IDNs and universal acceptance, a bit of emoji issues and the SSAC guidance around emoji domains, as well as IDN homograph attacks. We've also started to incorporate some of the material from the last ICANN meeting in Kobe and the DNS symposium in Bangkok into our training materials and guidance. There is another set of workshops and trainings that we do around the subject of DNS abuse and misuse. We do this for the operational security community and law enforcement. This tends to be around how to identify threats and challenges, so this is an overview of where we've gone during this year.

In February, ICANN collaborated with the Pakistan Telecommunication Authority to deliver a multi-day workshop and that covered a variety of subjects around building the capability in the country, so that their law enforcement ISPs and others could understand what the latest threats and trends were, how could they recognize these issues, and then help them spread the awareness in country. In April, we delivered some similar workshops in two locations in Lebanon, and that's just an example of the types of training that we're doing.

As I mentioned, we're also doing training with law enforcement agencies, we've been collaborating for many years with Interpol and Europol and this is an example from earlier this year in Korea. We do this training for the law enforcement community because it's one of the objectives that came to ICANN through the GACS Public Safety Working Group. They have been supportive of developing DNS abuse mitigation capabilities for law enforcement authorities, increasing the participation of law enforcement organizations in ICANN and raising the awareness of who are the proper points of contact for law enforcement to reach out to registrars and registries when there is an issue.

Now, how do we show some impact of these DNS abuse trainings? Well, if you look back over 10 years ago, we had quite a bit of good community collaboration that came out of responding to the [inaudible], more recently around the Avalanche and Andromeda domain generation algorithm takedowns, a number or registries that have been using the expedited registry security request process or different individuals coming to ICANN using the coordinated vulnerability disclosure process. So, these are just some examples, it's not directly related to our abuse training, but it does in some cases help inform decision makers of who they need to talk to when they're trying to deal with a cyber attack or some type of incident. And perhaps the lack of queries getting misdirected is maybe one example of how people are starting to learn who they need to approach and streamlining the process of communication when there is some coordination needed.

Another example that is quite timely, of a different type of training, has been the registry and registrar training. So just prior to this meeting, ICANN Staff and participants from the registrar stakeholder group delivered a number of trainings, not only in Lisbon, Portugal, but in Kampala and in Moscow. This is a program that has grown out of collaboration with the registrar stakeholder group and it seems to be quite successful. It's aimed at training the registrar staff that are either newly accredited by ICANN or that are interested in becoming accredited registrars. Also, the registrars are encouraging their newly hired staff to go through this training, so particularly those that are involved in the compliance processes and policy work.

The workshops help cover operational issues and concerns the registries and registrars might have, and it helps I think strengthen the awareness of who in the registries and registrars needs to contact ICANN and the different stakeholders when there are issues. So this one final plug, if your registry, registrar or your government is interested in these types of trainings, I encourage you to reach out to your regional global stakeholder engagement representative at ICANN and then they will filter the request up to our office of the CTO team; that way we don't have multiple queries coming in through different parts of the community to different parts of the organization and we can have a streamlined process to manage this interest. So, with that, hopefully there is time for questions. Thank you.

EBERHARD LISSE: Yes, thank you very much, short and sweet. There is time for questions. I always ask myself whether the number of workshops presented is really a good indicator of the impact achieved. What change has this effected. Can you imagine that? Actual change?

PATRICK JONES: I think in some cases I tried to show the stat about the ccTLDs starting to sign their zones. Another indication is that there is not misdirected or misguided queries from governments or law enforcement agencies, that now there is particularly after we've delivered a training in those locations, the regulators and the ISPs and others know who their contact points are, not just at ICANN or at their registry or local registrar. And we don’t really have good data on that right now, but I think the sense is that after we deliver training, there is a positive aspect to it in those locations.

EBERHARD LISSE: Thank you very much. I take this with particular interest to hear that it is often helpful for local government to contact ICANN to find out who their local counterparts are, but that's not unusual. Anyway, thank you very much. We are running a little bit ahead of ourselves, which is good. Our next presenter is Jay Paudyal. He is a fellow, and we have a longstanding tradition, though not always managed to fulfill it, and we managed in Kobe, and we manage now to invite fellows to present here. And I always tell them that we are a very friendly audience. So, if there are questions, be respectful of the fact that he is a fellow and he may not be as experienced to us nerds than others would be.

JAY PAUDYAL: Thank you for saving me. Good evening everybody, I am Jay Paudyal, member of Neo-Brahmi Generation Panel and a fellow from India this time. Today, my talk is about IDN and its scope and challenges. What are IDNs? They are Internationalized Domain Names. I have 12 slides to speak. You have seen demand for use of regional language content in their own language, internet users. I believe who knows English are already on. I have seen a survey which says 90% non-English speakers. So, there is IDNs because in some areas people don't understand Latin set, we have mobile phones and we can type as in spelling. But for example one the regional content, so IDNs might engender some fit.

We have some challenges with IDNs because when internet was born, it didn't seem like it will reach to Facebook, we will have WhatsApp. What kind of challenges they are? Main challenge, I think, awareness is the main challenge and if we talk about technical challenge, we have so many organizations working on it, we can solve technical challenge by doing some technical stuff, but one by one I have seen most of the popular websites don't recognize IDN names or username, or email ID. If I go to Facebook, it won't happen. And a few browsers don't redirect IDNs to proper URL, which will lead to confusion for end user. And there is searchability issue as well, but big giants like

Google, Amazon and other players are solving this issue and they are working heavily on ideas, as well.

I own a couple of IDNs in Devanagari script. I have seen some registrars don’t support redirecting or forwarding of a domain name to another domain name, suppose if I have, you know, singular and plural version of domain name, I can't, you know, redirect or forward simply, I have to put Unicode off that string and normal domain name owners or end users, they are not aware of this thing. Technically I can Google and convert it to Unicode. Linkification is used in popular applications, like if I type any IDNs, URLs in popular applications, it will not convert into a clickable link and always phishing and spoofed URL an issue with IDNs. If I show you some example, these are IDN domains. See, A and B are looking similar, but they are not.

Like, if I can read hindi.com and hindi.com, I have put a little dot ahead of that word, but it is looking similar. You can find a difference here, but what is an end user type in that small address bar, they might miss it. Second, it is canar.com, sound-wise, and B is ravana.com. They are looking similar. If I do it with some fonts or converter has some different font setting, it will look similar. And the last one is Monday.com, and Monday.com. see, both are correct, they are just different style of writing a word. If I ask someone to open Monday.com, they might open B, if I'm talking about B, they might open A, both are correct. I know I'm showing you these URLs are challenges, but in fact, they are not, because we have this kind of problem in ASCII domains, as well. Can you see, apple.com, and B, the second one is not 'l' and it's not the case of, you know, cross scripting, where we mix two scripts. I have used Latin script in this. That's not L, that's J, and you can see Orange.com, I can put zero in and with this font is looking a bit different, but if we are on different font, it might look similar. Google.com, again, they both are different words and Wilson.com, "W" traditional one, you can put double V and make a "W".

So, solutions. ICANN gave birth to IDN and challenges as well. These days we have UASG, Universal Acceptance Steering Group, which is solving this issues. They are making some standards, they have just announced their new action plan, what they're going to do in the next couple of years to create awareness, to create technical awareness between developers and I request you to join this group, uasg.tech. And second one, awareness is the key. If we are content publishers or if we are developers we can do awareness, even if we are registry or registrar, we can do awareness program for IDN domains, and it should be more compatible with web browsers and applications and we need support from registries and registrars, as well, as I just mentioned.

I cannot redirect or forward IDN to different IDN by putting the same in Unicode, I have to convert for string in Unicode, and there are so many tech giants are already working on IDN indexing and searchability issue. And see, to solve this problem, to solve this challenge, there is one mantra, Accept, Validate, Store, Process, and Display Unicode Data. That is looking simple, but you know, in practical it is quite tough, because we need to follow UASG guidelines in every software application, in every website, and every e-commerce, but we have to do that, because it's all about money, it's all about driving a market. If there is a demand, we need to fulfill it. ICANN is based on multi-stakeholder model and we need support from this model as well, like registries need to provision, resolve and manage IDNs and raise awareness about it. Registrars can distribute, can sell domain names heavily and can do net marketing or traditional marketing heavily to promote those names and content publisher, they can create more content using IDN domains, as well, and application developers should follow guidelines of UASG and obviously last but not least, governments. They should provide such infrastructures and they should create some standards or do some developments around some policies to support IDNs heavily. And we should not forget, it's always about the end user. That's all from me. Any questions?

EBERHARD LISSE: Thank you very much. That was quite interesting. We hear this regularly, last time in Kobe we had a presentation from Thailand, same thing. It's positive that you bring this, because it shows the need, and we had this in Shanghai, in Beijing at the ICANN meeting. What happens, what usually happens, the presenter comes, my mother-in-law, she got a cell phone, she got connectivity, she can do the WhatsApp in her own language, but it's very difficult. My mother-in-law doesn't speak English, she speaks a language called Oshiwambo, but it uses Roman letters, so it's not a problem. She can WhatsApp with my wife, she can figure out one of the newspapers and go to the Oshiwambo section, but if you speak Thai and you only write Thai language, if you speak one of the major Chines languages and you only write that script, or one of the Indian languages, it really makes it, you have a cell phone and you're still not able to connect. So I fully appreciate that any effort and support efforts that are being made to reach more people who have access to the technology, but who need to use the language.

CAMERON BOARDMAN: Thank you, Eberhard, Cameron Boardman from .au. We're thinking about introducing IDNs into our namespace, can I ask the first question as a proportion of your total names under management, how many actual names registered as IDNs do you actually have?

JAY PAUDYAL: Can you come again?

CAMERON BOARDMAN: How many IDNs do you currently have registered in your registry?

JAY PAUDYAL: My job, my day job is to build software and to build e-commerce websites, and I have booked a couple of IDNs, I think the number is 10, and two websites are already live, for news and e-commerce venture. What we are doing is we are developing, see, I'm not just talking here to promote IDNs, we are developing [inaudible] as one of the biggest e-commerce of India and we are planning to promote, you know, in rural areas of India. And see, what we want to show, we want to prove that it has business potential, so that people can use it.

EBERHARD LISSE: He's not from a registry, so he doesn't have, that's the question that you were having.

CAMERON BOARDMAN: I was asking a more general question, let me ask him in a more general question. Are you able to make any commentary around whether or not there has been an increase in spam emails associated with IDNs under any namespaces that you might be familiar with?

JAY PAUDYAL: The spam issue is problem with every skips domain.

CAMERON BOARDMAN: But is there any difference to your knowledge around spam associated with IDNs?

JAY PAUDYAL: You're talking about spam, IDNs?

UNKNOWN SPEAKER: Can I comment on that? Apropos your two questions, last tech day I surveyed all the ICANN contracted domains and there were about 2 million IDNs in the contracted domains, which is between 1% and 2% of the total. Half of them are in .com, half of them are spread all over everything else. There are some new TLDs that are IDN only, but they are quite small. And in response to your question about spam, the number of mail systems that actively accept EAI mail addresses is actually quite small, so that even if it were totally spam, which it isn't, it would be a rounding error in the gush of spam that we see everywhere else.

CAMERON BOARDMAN: That's very helpful, thank you.

EBERHARD LISSE: Alright, take your time. I think I'm also going to have to call for a break after this and then we do the long presentation after a short break of 20 minutes.

ABDALMONEM GALILA: Yes, Abdalmonem Galila, Universal Acceptance Ambassador. Awareness is not only for the end user. Awareness is also for service provider. For example for anti-spam. If anti-spam service provider knows how to make an application and spend energy to validate the domain name or email address correctly, it will be easy. And other comment, I agree with you totally that awareness is the key, but at same time, if you want to use IDN and use EAI at the same time, you as a service provider, your application is universally accepted and validated, comparing with the [inaudible] universal acceptance, it is useless. So it is ecosystem, end user, and service provider. Thank you.

EBERHARD LISSE: I feel that the more often we bring this topic up, the more people we reach on whatever level, registries, registrars, content providers, end users. It's easy for us techies to say if you can't read an LFC which is written in English, bad luck, but end users who don't speak English, which is the majority of the end users in the world, don't do that.

ABDALMONEM GALILA: Actually when we saw .com, there are also hesitation using .com, so people hesitate new changes. It's us who should push if it is for right thing. And market will drive everything. Thank you.

EBERHARD LISSE: Okay, thank you very much. I think we can give him another hand. Anyway, I say we do 15 minutes and that will mean we can be right on time with the next presentation. [AUDIO BREAK]

Okay, the few people still standing, please settle and sit down. We are now going to have John Levine teach us all about DKIM and Superficial and all the other things that I don't understand and don't work.

JOHN LEVINE: Well, I must say I am honored and touched to have such a wonderful introduction. I'm talking about mail security in the DNS and there is a great deal to say about mail security. But since we only have limited time and this is ICANN, I'm only talking about the parts of it that are related to the DNS. An important thing to remember about mail compared to pretty much anything else we use, is that email is really, really old, it is one of first applications that was on the Arpanet, the internet's predecessor, and it reached its current form really early.

As my slides here say, there is a document from 1977 that describes the format of email messages, and if you look at that document, the format now is essentially the same, I mean, we've added some stuff to it, anything that met the spec, anything that matched that spec would pretty much be a valid message now.

SMPT which is the protocol that us used to communicate from one mail server to another was first documented in 1981, and again, if you look at that version of SMTP, we have added some new features and we've deleted a few old ones that nobody used. But it's essentially the same. And so the DNS wasn't invented until two years later. The first DNS spec was published in 1983 and the MX record which links mail to the DNS wasn’t published until 1986, so mail had been going practically for a decade before it was lashed up to the DNS. And before the MX record, there was some early experimental records that didn't work, but even in 1986 when you look at that RFC it talks about the transition plan from the old previous host file version of addressing to the DNS.

So everything that is related to the DNS and email has been grafted onto an existing system and it's been grafted onto an existing system that runs 24 hours a day, so people have likened it to open heart surgery. You can't stop the mail or the patient will die. Which has limited the sorts of things we can do and which has led to some compromises, some of which are good and some of which are bad. The other thing I want to mention about mail is that nobody cares about your mail as much as you do. You send the mail out and it may be received. If it's a personal mail from my wife, I care about it a lot. If it's a generic advertising mail from some airline, I basically don’t care at all whether I get it or not. But if the sender cares deeply, when you say why like, why don’t they deliver my mail, the basic answer is because they don’t care.

Ah, I see you didn't use my version of the slides. If this were scaled down a little better, this is intended to be a picture of the internal mail architecture and on the left we have somebody sending a message and she's using a mail user agent which is either a program like Outlook, or it's web mail, which then sends it off to the sending MTA, mail transfer agent, which is mail server on her mail system, which then sends it over the internet, and that middle arrow is supposed to be on top of the cloud, which is supposed to be the internet, to the receiver's mail transfer agent. So it goes from the sender to the sender's mail system, to the recipient's mail system, and then it goes down, and what you can't see on the right there is the recipient also has a mail user agent, which again is a program or it might be webmail, and that's how the recipient receives it.

Since this is the internet and this is tech day, we're all over three letter acronyms, and the internet has a rich and varied set of them. The sending program is a mail user agent, an MUA, which then submits the message to the first stage in mail processing, which is a mild submission agent, and the difference between a mail submission and a mail transfer agent is essentially the mail submission agent will take mail from all of its own users, it won't take mail from the outside, and the MSA also tends to do validation and cleanup, so for example if the message doesn't have a date or doesn't have a message ID, the mail submission agent will add it. It then passes it onto the mail transfer agent which is typically on the same computer, sends it over the internet using SMTP to the recipient's MTA, which then sticks it in a mailbox.

Then the recipients MUA will then have to pick up the mail somehow and again it might be webmail, or there are two systems called POP and IMAP which are used for the recipient's mail program to pick up the mail from the recipient MTA. So it's these three steps. It's submission, there's SMTP which actually might be more than one step if it goes through more than one mail server, and then there's POP or IMAP to pick it up.

So, what problem are we solving? Back in the good old days, back when mail was invented, every mail message was a wanted mail message. There was one spam in 1978 which was world famous, because it was the only one for the entire decade. That stopped being true in the 1990s. In the mid 1990s we started getting spam through various ways. At that point spam just tended to be annoying, it wasn't dangerous. It was ads, there was a famous spam for an American Visa lottery, there was a guy who was purported to be sending out copies of the Hiroshima atomic bomb, which he got out of the American Archive somehow. And in recent decades, then spam has gotten significantly worse because it's not just annoying, it will contain malware, it will contain phishes that attempt to steal your identity, so it has become actually dangerous.

So originally filtering out spam was mostly a convenience so you could find the good mail among the bad stuff. Now it's essentially. It's going to keep malicious stuff out of your mailbox so that people aren’t scared to open their mail. Spam filters are very complicated, and again, I am not going to talk about all the ways that spam filters work, I'm just going to talk about the parts that relate to the DNS.

So, for DNS-based spam filtering, we have a couple of handles that the recipient system can use to figure out whether it wants the message. And what we have here is a stylized MSTP session where it's sending a message from a sending system to a receiving system. So, the first thing that happens is that the sender connects to the recipients mail server, so the recipient now knows what IP address the message is coming from, so the first line there, the 220, that's the recipient server saying yeah, I'm here.

Then the next thing it does is that the sender sends a hello command with what it claims is the domain name of that mail server, which if it's a real mail server, it's likely to be true, and if it's not a mail server, if it's spamware, it's likely to lie. Then it sends what we call the envelope, which is the mail From: address, which is the minimal sending address, which in this context is address to which error reports should be sent. It may or may not match the address on the From: line. And there is a recipient address, which is who do you actually want to send the mail to, which may or may not match the address on the To: line. And typical cases where they would disagree are, for example if I send a message to two different people on two different mail systems, the To: line will have both of their addresses, but it will send separate copies to the two mail systems, so each mail system, the envelope will only have the recipient on that mail system. And for the mail From: address, if it's a mailing list, the mail From: address will actually be an address of the mailing list, so in case the messages bounces, the list can handle it.

One thing that is also possible, you will notice in the middle there, it says mail From: and then angled brackets with nothing between them. It's possible to send a message with an empty return address. This is typically used for status messages to avoid mail loops. So if you send a message and then the recipient says no, I'm not going to accept it, and the recipient then sends a bounce back, you don't want the bounce to bounce, so the mail From: with an empty address says if you can't deliver this, just throw it away.

So, the first use of the DNS is how does the sender find the recipient, and that's what MX records are for. The sender knows it wants to send for example to bob@example.com, so it then looks up the MX record for example.com, and it finds an MX record that happens to be mx1.example.net, there's no need for the name of the mail server to match the name of the domain that stuff is being sent to, and these days typically they will have no relationship to each other. For example, there are probably a million different domains that all get mail hosted at Google and their mail servers all have Google.com mail server names. And then the 10 there, as we saw, if you saw Wes' talk this morning, the 10 says what the priority of this is. There could be multiple mail servers and the priority says what order to try them.

So, once you have the name of the mail server, then you have to go back and say well what's its IP address, what's the address of mx1.example.net, and you will get back A records or AAAA records and those actually tell you what mail server to connect to. Now it is also possible for a domain to have mail but not to have an MX record. This is a compatibility feature left over from the 1980s. If there is no MX record, then you pretend there was an MX record that has the same name as the domain itself, and you skip ahead to the A and AAAA records. And this is an example of how nothing in email ever goes away. This is a feature that was installed around 1986, and it really should have been taken out around 1990 when everybody already had MX records, but we're still stuck with it.

The next use of the DNS that people typically do, is they do a little sniffing to validate the IP address from which the message is being sent. And the reason for this is that on the internet these days the vast majority of computers attached to the net are not mail servers, they're phones, they're PCs, they're home users, and they are sort of random hosts at hosting providers. And so one question is like, is the system sending this mail a system that is plausibly a mail server? Because you don't want to accept mail directly from somebody's phone. The phone should use submission, which I showed you before, to send it to the provider's mail server, and it will then be sending the mail.

So, what you do is you do a reverse DNS lookup, what's the name that goes with this IP address, and in this case it's mailout.example.net and then you flip around and do a forward lookup and say okay, what's the IP address that matches this name? And if you get a valid name and then the name resolves back to the IP address you start with, that tells you this is a statically assigned IP address, it's not a dynamic address that is going to be assigned to one phone today, and another phone tomorrow, and it is plausibly a mail server. And the other thing that people check is does it have a name that looks like it might be a server, or does it have a name like this one to the lower right, that is a host on a broadband provider in New York City, and that's the actual IP address of his home. He did not send me mail directly from that address, he sent it through his provider, but if he had tried to send it from that address my server would have looked at it and said nah, that's not a plausible sender. So the PTR validation is the first use of the DNS here. And this is a heuristic, but it's a really reliable one. If this passes, the chances that it is a real server are very good. It might not be a nice server, but it's a real server. And if it fails, the chances that it could have sent you legitimate mail are extremely low.

So the next part you go to, this is the most contentious part. There are people who publish black lists and white lists of Ips and domain names. And this is their opinion about like you do or do not want to accept mail from this provider. I have commented that any moron can run a DNS black list, and many morons do. You can find a multi-RBL wegpage that will show you here is what 100 black lists say about your IP address, and you'll probably find like wow, I'm listed on 20 of them, and you know what? You don’t care. Because it turns out the number of blacklisted people actually use is like three; I mean, there's SpamHouse, Trend Microsystems does one, and there are maybe one or two others. Anyway, the way DNS black lists and white lists work is they basically use a mutated version of reverse DNS. You take the IP address, you reverse it, you prefix to the name of the black list, which in this case is a fake black list bl.badguys.net and you look it up. And if you get no answer, that means it's not listed, which if it's a black list, is good. And if you do get an answer, you will get an A record like this one, 127.0.0.5

which says that the black list lists this IP address. And the low order bits usually are supposed to give you a hint of why it's listed. Listing criteria range from I got a lot of spam from this to this is an address at a residential provider that said that their users aren't supposed to be sending mail directly, and then there are silly reasons too, like, I tried sending email to this postmaster and it bounced. For usage of black lists, some of them are reliable enough that you can use them to block mail outright, which I do. A lot of them are sort of advisory, so you look up to see if something is in a black list, and if it is, you use that as part of what you do to compute a spam score. There are also DNS white lists, they're not very interesting. I spent half a year attempting to set up a white list for the SpamHouse project which runs the biggest black list and we mostly discovered the people whose mail we were willing to white list didn't care, didn't want to be on the white list because their mail was already being accepted fine, and the people who did want to be on the white list were the people who shouldn’t be on the white list, because people were rejecting their mail for good reasons. So, other than very localized things, like these are the IP addresses of people in your same company, or something like that, we use black lists a lot, we don’t use white lists at all.

Beyond that, you can use the same technique to check domains. And the domains you are most likely to check are the ones in the envelope. It turns out the even if spamware lies about the domain in the hello line, they tend to lie in a consistent way. People also, when they're doing body checks of the contents of the email, they will go through and look at the URLs and they will look at the domain names. And there are frequently domains that are just set up for a few hours or a few days to send out malware and stuff. So again, you take the domain that you're not sure about, in this case, maybe.org, and you look it up in the black list, called dbl.badguys.net, and again, if it’s not listed, you will get no answer, and if it listed you will get something about why it's bad.

And depending on the reasons, you might give it a higher or lower score. For example there's a widely used list that lists domains registered within the past day or two, which are disproportionately likely to be malicious, as opposed to domains that have been around for a long time, or there are domains that seem to be sending phish, well, you don’t want to get mail from them. So, again, for the checks in the envelope, if it's a phish domain and if it's the hello address or if it's the from address, you can block it reliably and not lose any real mail, otherwise you take the score and again you add it in to do the body spam scoring.

Now the next three bits are going to be about how we validate mail, which is, is this mail from who it purports to be from? And again, back in the good old days when every mail was nice, nobody sent fake mail, or on the rare cases when they did, it was an April Fool's joke. Now, if you're a spammer, yeah, everybody sends fake mail and just by total volume, the vast majority of mail is spam and the vast majority of the spam lies about where it's from. So if you can come up with some reliable identifier for where your mail actually came from, and you cam recognize senders who have historically behaved themselves, that is a good way of identifying mail that you want and accepting it. So, what SPF does, it does what we call path validation. SPF says mail form this domain, and again the domain is the one in the mail From: line. In this case it's example.com or if it's an empty bounce thing, then you use the domain in the hello, which in this case is mailout.example.net.

So for SPF, you take the domain name and you look up a text record, and the text record comes back in this fairly complicated format, v=spf1 mx ip4:203.0.113.0/25 ~all, and SPF was sort of over-designed to make it very easy for mail senders to publish SPF without making any changes to their mail architecture at all. So the SPF record has a whole bunch of different ways that you can describe where mail from this domain name can legitimately come from. So, in this case, the first thing is MX, any host that has an MX record pointing to it for this domain, that's a valid place for it to come from. You can specifically list specific ranges of IP addresses, so anything from that /25 of IP4 addresses is okay, and the last thing says ~all, what we call a soft fail, it's like it's probably not good, but I'm not making any promises. So, SPF, this is either going to be really easy to send up, all you have to do is accomplish one text record, once you figured out who sends your mail, and the results are squishy, it can range from no to no opinion, to soft, it can be fail, soft fail, indeterminate, or valid.

So, where we actually use SPF is it's partly used to white list sender who you know are friends of yours. Partly it's used in DMR which I'm going to discuss a few slides ahead here. SPF is quite limited in the kind of mail it can describe. So for example, my wife has an email address at her university and if somebody sends her mail at her university address, it goes to the university and then they remail it to me. So when I see mail, no matter who was originally from, it came from a university. So the SPF record will describe the IP addresses of the original sender, but we're not getting it from the university, so from that forward the SPF will always say fail, which more often than not is wrong, because most of what the university forwards is actually real mail. Yeah, Barrett?

THOMAS BARRETT: This is Barrett, just a side note that my university is just terminating their forwarding service because this doesn't work. That was their decision, too many misfires, and they just decided not to do it anymore.

JOHN LEVINE: My university hosted it at Google so it's not their problem. My wife's university hosted it at Microsoft, so it's not their problem either. And the final point is even if SPF comes back and says, yessiree, this message is super duper, clean, perfect, 100% validated, all that means is the SPF passed, it doesn't mean that the mail is any good, because spammers can publish SPF just like bad guys can and we discover that historically spammers tend to be early adopters of mail validation. Ignoring a lot of botnet junk, but closer to the medium quality spam, it tends to validate at slightly higher levels than legitimate mail. So, all it means is that at least as far as the envelope of the message is concerned, it was actually sent by the purported sender if SPF passes.

Our next attempt to solve this problem is DKIM, which does not validate the path, it validates the actual contents of the message. So, the way DKIM works is that first makes a cryptographic hash of the body of the message and then it makes a cryptographic hash of the headers of the message and then it puts all those hashes with some other stuff into a new header that it then adds to the message. So the thing in the middle that says DKIM-Signature, that is a typical DKIM header that would be added to an outgoing message. The interesting parts, the d=, that's the domain that is responsible for the message, that's the domain name that is signing the message.

There is also a thing called a selector, which in this case, my selectors are time based, s=k1906, so that you take the selector and the domain name and stick the word domain key in between just to avoid collisions. And use that to look up a text record which will have a cryptographic validation key that the recipient can then use to go back and validate the signature. So, the recipient then, when it receives the message, it then recomputes the body hash and sees if it's the same hash, and if it is, it then recomputes the header hash and sees if it's the same hash and then if it is, it goes back and does the cryptographic signature validation, and if the signature validates, then congratulations, this is a valid DKIM signature from example.com.

Another difference between DKIM and SPF is it is fairly common to have multiple DKIM signatures on the same message. So, for example, my mail system hosts a whole bunch of domains for a whole bunch of other users. So if mail goes out in one of my user's domains it will have a DKIM signature from that user's domain and it will also have a DKIM signature from my own mail server domain. So recipients can say that it's from my user and it's from mail system, since we are both to some extent to blame for whatever is in the message.

So again, if these things pass, these give you domain identities to attach to the message, it shows the message is good, because again, bad guys can put DKIM signatures on their messages too, and again, like SPF, it's typically used to white list known friendly domains and it gets through and into DMARC. It works better than SPF does for forwarding like when my wife's university forwards the mail, it generally does so without reformatting it, so the DKIM signature remains valid. On the other hand, there are forwarders like mailing lists, that will add footers, which will break the body, it will add a tag to the subject line, which will break the header hash, so DKIM is far from a panacea. And when we designed DKIM we had this in mind. If a message is to some extent your responsibility you should sign it.

So the mailing list may break the previous DKIM signatures, but it puts the list's own signature on it. So you can say there's a valid signature from the list because it was added after it was reformatted, so I know this is really from this mailing list and that this mailing list is a real mailing list that my users like, I should probably deliver it. So, to answer your question about why Google doesn't accept your mail, Google has a policy which they are big enough that they can enforce, that if you're sending mail over IPV6 it must pass either SPF or DKIM, or they won't accept it.

And the theory is, it's like if you are technically sophisticated enough to turn on IPV6, you're probably technically sophisticated enough to at least add an SPF record, since that literally is one text record in the DNS. And your mail system may well also be smart enough to do DKIM; mine is. So we had SPF and we had DKIM, both of which are essentially whitelisting systems, both are ways to say this really is related to this domain and if you look up the domain it says yeah, this domain actually sends good stuff, they you'll probably want to deliver it. Otherwise, you do whatever kind of filtering you want. DMARC then takes these same technologies and flips them around and it was originally designed for domains like PayPal.com, which is phished stupendously heavily and PayPal's mail, one thing that's very unusual about PayPal's mail is that it's very unimportant.

No matter what message PayPal sends you, it's always the same thing which is go log into the website and see what happened. So, if the message gets lost in this particular instance, you're not actually losing anything of importance, because the user can always go and log into the website, and since they know who is logged into the website, if the message got lost and you don't log in for a while, they can send you another message that says, hey, you have a message waiting. So, PayPal and their friends said we want a way to kind of explain that PayPal mail is so heavily phished and so unimportant that it's worth taking the risk to throw it away if it's unauthenticated, and that's what DMARC says. People will claim that DMARC says other stuff, but that's what it actually says.

So, what DMARC does, you start with the domain name in the From: line in the header of the message, not the envelope, but the actual From: line. So this message is From: Mr. Bob <bob@example.com>. So you check it, is this message DMARC valid for example.com. so, first you look at the SPF. If the SPF passes, remember SPF has fail, soft pass, it has to be a full pass, and the domain has to be example.com or a sub-domain. So if that's the case, then it's an SPF pass, and that is what DMARC calls aligned, if it's example.com and the From: line is example.com. Failing that, you look at all the DKIM signatures. If there is a valid DKIM signature, it has an d=example.com, then that's aligned. So if either of those are true, if either the SPF passes and with the right domain or there is a DKIM header that passes with the right domain, then the message is aligned, and if the message is aligned, DMARC says that's great, this is a wonderful message, as far as we're concerned, we're done, go ahead and deliver it.

And so here is a typical DMARC record. The way you find the DMARC policy is you prefix _dmarc to the domain name and you look it up, and there is long bunch of stuff here, but the one that is most relevant here is p=none, which is what my DMARC records say, this is the sender's policy, what it recommends you do. If you get a message that purports to be from that domain but is not DMARC aligned. And the three possibilities are none, which is do whatever you do otherwise; quarantine, which says stick it in the spam folder; and reject, which means bounce it. And for PayPal and organizations like that, it clearly makes sense to say p=reject, because again, if you lose the mail, it doesn't matter. For mail domains that are from normal people like you and me, a policy of none is probably more likely, since you are not personally a major phish target and with mailing lists and forwarders and stuff like that, there is actually a certain amount of legitimate mail that is real mail, but fails DMARC.

Unfortunately, although it was originally intended for highly phished targets like banks and PayPal, a few years ago, AOL and Yahoo each had huge security breaches and essentially all of their user address books got stolen, which meant you now had Yahoo users getting spam that appeared to be from other Yahoo users who they knew, because the spammers had stolen the address book so they know that this Yahoo address is in this other Yahoo user's address book. So AOL and Yahoo were getting huge numbers of complaints, like why is my friend sending me spam, and the answer is she isn't, but then it goes into technical stuff which to the recipients sounded like gargle, gargle, blah, blah, your fault, can't help, goodbye.

So since they hate getting support calls, they turned on p=reject since all the spam was coming from places outside Yahoo or outside AOL, that suddenly made all of that particular spam go away, which was good for them. The bad news was that there was a small amount of high value mail, like mailing lists, which as soon as AOL and Yahoo turned this on, every Yahoo and AOL subscriber mailing list, their mail started bouncing like crazy. There are a lot of DMARC cheerleaders who keep insisting that everybody should publish p=reject, just because it is more secure. Which, if you're a business or a bank and you have full control over all the mail your users send, that may be good advice.

If you're an ISP or if you're a provider with normal mail users, it's not such great advice, because they're going to lose real mail. And it's not just mailing lists, but that's by far the most visible thing. And we had long, long advices. People were saying it only affects 1% of the mail, why do you care? And the answer is because it's 1% that we care about. There's 73% of mail that's DMARC validated, but it's all bulk commercial mail that the recipients don't care about, whereas the 1% of mailing lists, we do care about that.

So, before I tell you about how we fix this, here's how we analyze it. The other clever thing that DMARC invented was it has this reporting feature. There are what are called aggregate reports and failure reports. So, in my case, I send all my aggregate and failure reports to an address, an aggregate report is what it sounds like. Google sends aggregate reports, they'll take all the mail that reports to be from my domain and they'll make a nice XML thing that describes all the mail they got that purported to be from my domain, and they'll put it in a message and mail it to me, and they do this every day, as do lots of other places.

So I now have 100,000 aggregate reports that I've collected over the past 4-5 years, so I have a pretty good idea of what mail appears to be coming from my mail system, which is useful both to see who is faking my mail, and I can find some interesting things, like I know how many people at many large providers subscribe to the nanog mailing list, because as soon as I send mail to NANOG which then mails it out to, say, Google, at which point 5000 people at Google now get my mail which doesn't pass DMARC because it added a tag of some sort. But if you're also doing commercial mail, that way you can make sure that you don't have third part senders sending on your behalf. There are also failure reports where you say if a mail message shows up that is not DMARC aligned, just send it back to me so I can see what's wrong.

So, in practice, lots of people send aggregate report, almost nobody sends failure reports because failure reports contain actual messages, and depending on how things are screwed up, the actual message may not really have come from the person in the domain line so basically you're sending a message from one party to somebody who may have nothing to do with it. So, in practice, for reason, LinkedIn sends failure reports even though they belong to Microsoft, and I get a lot of failure reports from Chinese ISPs, which are invariably random Chinese spammers who thought it was funny to fake abuse in one of their spams. But this allows you to analyze what's going on and see exactly what's failing. So I know in painful detail, whenever I send stuff to a mailing list, I know all the places the mailing list sent the mail and all the places that the DMARC failed, which would have rejected my mail if I had a policy.

So, after a couple of years of people complaining of mailing lists, the people who brought us DMARC then came up with something called ARC, which is supposed to fix the damage that DMARC causes. A friend of mine said, "They sold us a fountain pen which leaks, and now they're selling us a rubber glove to wear while we're using the fountain pen, so the ink doesn't get on our hand."

The point of ARC is supposed to be that as mail gets forwarded from one site to another, each forwarding system puts what's called an ARC seal on it, which essentially describes the validation status of the message when it got to that system. And then the recipient looks back through the sequence of ARC seals, it can see where the message came from and what happened to it, and even if the DMARC is invalid when it arrives, it can sometimes look back and say, oh I see what happened, it's okay anyway.

So, I'm not going to go through this in detail, but this is what an ARC seal looks like the first line is ARC-Seal which is the signature for the other two lines. The second one is ARC-Message-Signature, which is basically the same as a DKIM signature, but has that i=1, since the first seal on the message is

#1, and the second is #2, and so forth. And the third is ARC-Authentication-Results, which tells you at the time you received the message, in this case it says for the message I'm describing here, it says what IP it came from, there is said the SPF value works, but did not have a DKIM signature, but since it had valid SPF, it passes anyway, that's why it says dmarc=pass on the bottom.

So the idea is that the recipient will can check the chain and go back and say well if was valid in the first place, then it was probably okay. Now the next thing to say about ARC is, you know, if I'm a bad guy, why don’t I put some fake ARC seals and say oh yeah, this message was great. Can the final recipient go back and validate that the ARC seals are in fact real? And the answer turns out to be usually, but not always. It only really makes sense to look at ARC seals if the message is coming from someone who is generally trustworthy, which turns out to be okay, it turns out the number of systems that actually send mail from mailing lists is small enough that most mail systems know pretty much who their mailing list senders are.

So, then I asked some people at a large sender, if you're only going to look at the ARC seals that are on mail coming from nice mailing lists, why don’t you just whitelist it all? And they said the reason is because a lot of mailing lists actually do pretty bad spam filtering, and frequently the only filtering they do is to check that the sender address is somebody who subscribes to the list. So, in fact, like on the SSAC list, we're constantly getting spam from one of our members whose mail is faked. So the point here is that the ARC retroactively will go back and say if the mailing list had been the kind of filtering that we do, would it have accepted the message, and they go back and say, oh, like this one, originally the DMARC was okay, so they should have accepted it, whereas that one, it wasn't, so it's not.

So ARC is a nascent technology, the code is pretty much written, the libraries are pretty much written. Google's mailing lists apply ARC seals, my mailing lists apply ARC seals, it's sort of implemented at Google, it's sort of implemented at Yahoo and AOL, which are now the same company. And so at some point when ARC looks a little better, we should be able to have our mailing lists working the way they were before, but it is definitely a work in progress and this is I think one of the best examples of hack upon hack, we have ARC piled on top of DMARC, piled on top of DKIM, piled on top of SPF, piled on top of the DNS, piled on top of mail that goes back to the 1970s. But again, with email, this is the kind of stuff we have to do. Because if we were designing it now, we would do it differently, but we aren’t, and it really has to be backwards compatible.

Wes talked about this, this morning, and I'm going to talk about it now, since not everybody was here this morning. If your DNS has DNSSEC, since we're here at ICANN, all of our DNS has DNSSEC, you can publish a key in the DNS that says this is the key that my webserver is supposed to have for TLS connections, but it works equally well for mail. And the way it works, here is a slightly longer version of a mail session which is the sender says extended hello, which tells the recipient system to say tell me what features you support, and one of the features is called STARTTLS, which is yeah, I can do TLS connections.

So the recipient says, okay, start TLS, do a TLS connection, at which point it then goes through the same kind of security handshake that a webserver does on an HTTPS connection, and as well as setting up a secure channel, it also means the server then says here is a certificate with my name in it, so now you know what name for the webserver you know what name the webserver purports to be, and now you know what the mail servers purports to be, and then after that the connection is encrypted. And the reason you care about this, whereas normally you would expect the sending MTA to connect directly to the recipient MTA, that doesn't always happen. One thing that happens fairly often, if the sending MTA is in some sort of data center run by a hosting service, if the hosting service, particularly if they're the kind of hosting service that has five dollar a month subscriptions that are set up automatically, spammers will try to send mail, so in fact what they will do is all outgoing mail will be forced through a proxy mail server that the hosting service attempts to use to filter stuff.

And sometimes we have hijacks, screwups, mistakes and stuff, and mail just goes to the wrong place. So what TLS lets us do is the sending MTA says, okay, I'm sending mail to mail1.example.com because that's what the MX record says. But in fact, when I do the STARTTLS the proxy comes back and says, here, myname is proxy. At which point the sender then knows that the mail has been hijacked, at which point typically it will then abandon the connection and either reject the message or try again later. But at least now for the first time it can validate that the mail is actually going to the server that it's supposed to go to, and it's been going to somebody else who might spy on it, or rewrite it, or do something else malicious to it. There is another system called MTASTS that does the whole thing without particularly involving the DNS so much, but if you care about that, you can travel back in time several hours and listen to Wes' talk this morning, which may or may not have been recorded.

So, here's what I just said, so the STARTTLS gets the security, the DANE TLAS says what the certificate is supposed to be, which can validate it, and either it validates in which it says yes, it's the right mail server, or it doesn't, in which case don’t send the mail. And I happen to know this works, at least on a few mail systems, because I set up my TLSA certificates and my wife promptly stopped getting mail from her mother, because her mother's mail is at Comcast, a large American ISP, which turns out to be an early adopter of DNSSEC technology, and I have screwed up my TLSA certificates, so they said, wrong server, no mail for you. So then I poked around and I got the right mail server, and now the mail continues to flow.

The last thing I'm talking about, for many purposes you want to divide DNS up into zones of control and we have the Mozilla Public Suffix List which is what everybody uses to do this. It's a horrible kludge, it is a text file that everybody downloads every once in a while, and it is a list of what it calls public suffix domains, and it's very useful, so we all use it. So, for example web browsers try to decide whether two domains are close enough that they can share cookies, and certificate authorities use it to figure out whether a wild card certificate is low enough for the DNS that it's valid, and DMARC uses it for what they call organizational domains.

So, for example, example.com may have a bunch of subdomains like sales.example.com and support.example.com, and since it is hard to publish DMARC certificates for all your subdomains, DMARC says if support.example.com doesn't have its own DMARC record, then find the organizational domain name, which in this case is example.com, and check its DMARC policy. So the example.com DMARC policy affects all the subdomains for example.com, which is good, because they're all part of the same company. We use the PSL for that. The DMARC spec says feel free to use something better, but at the moment there isn't anything better.

A question that has come up recently, these are public suffix domains, but we actually heard from two totally separate people. One guy works for the British government who said that anything under gov.uk is part of the British government so the British government gets to set its policy, so, even though gov.uk is a registry and there are some things registered underneath it. And a guy who works for .bank, actually now that was have domains, anything .bananarepublic, it's all one company, they all work for The Gap, so The Gap can publish policy for everything under .bananarepublic, because it's their TLD, they can do whatever they want with it. Also, .bank has strong contracts with its registrants which are all banks, and they say among other things that you must have strong DMARC policies.

So, there is an experimental DMARC extension that basically says if the organizational domain, if you don't find something there, look up one level to the public suffix domain which would be gov.uk, or it might be .bananarepublic or it might be .bank, and look the DMARC policy there, same DNS lookup, one level up, which allows .bank and gov.uk to publish very restricted DMARC policies for all of their subdomains. So, since the PSL we all agree is a kludge, can we do better? The IETF dbound working group went to some effort to try to come up with ways to do it. I put in one proposal which I thought was dandy, Casey Deccio who has spoken at a lot of ICANN meetings put in another proposal, at that point he was working for VeriSign, and we went around and around, but it turns out to be a hard problem to solve.

Because, if you want to put the boundary information in the DNS, well, these are the questions you want to ask, like, can people publish their own boundary information or does it have to be vetted and put in a separate place? Is there more than one kind of boundary, are the boundaries for DMARC the same as the boundaries for cookies, and how expensive are the lookups. Generally, if something is 10 levels deep in the DNS and you have to do 10 lookups, that's bad, because if I'm a bad guy I'll send out spam with a., b., c., dot z, and you'll have to do 26 lookups which will be a heavy load, so you need a way to do it and at least bound the number of DNS lookups.

So anyway, I thought my proposal was great, and I can tell you all about it later, but I won't tell you now, but it didn't get consensus in the IETF so we're all pretty sure that there are ways to publish boundary information in the DNS, we haven't figured out what they are yet. Once there is boundary information, then it will be useful for DMARC and it will also be useful for a lot of other stuff, for figuring out what range of domains are under the same control, so they have the same policy. And that is it. So, if I have time for questions, I will take some.

EBERHARD LISSE: Thank you very much. Two things, some homework will be required for me.

JOHN LEVINE: Publishing the SPF record shouldn't be too hard.

EBERHARD LISSE: I find it very difficult to send emails to my son when he uses gmail addresses, even if I don't have an attachment, it tells me I'm too stupid.

JOHN LEVINE: Yes, I say, publish your SPF record, you'll become smarter.

EBERHARD LISSE: It's published, it's not the point, it probably needs more study. What I would have actually really liked, to have some cookbook recipes what to set up on the server, that would have been sort of the icing on the cake. Really good presentation, don't feel criticized.

JOHN LEVINE: Actually there are some on the web, I'll try to collect some and send them around so you can pass them out.

EBERHARD LISSE: We'll do that. Any questions, please? This is a very difficult topic, even for experts, getting email right used to be easy, and it's now really, really, seriously difficult, and the big sort of the elephant in the market, Google, they just don't care, they do whatever they want and that's the way you have to adapt, and if you can't understand it, bad luck.

JOHN LEVINE: Well, but again, as I said on my first slide, only you care about your mail that much time. Google doesn't really need to me to defend them, but they get unbelievable amounts of spam, and they are defending their users from stuff so awful you can't begin to imagine.

EBERHARD LISSE: It's surely a matter of scale. I'm not saying Google is wrong, I'm just saying it needs some homework, you need to actually figure out exactly what to do and experiment until you get it right, okay.

JOHN LEVINE: And you're lucky that his mail is not hosted in Microsoft, because even people who work there say, I don't know. Barry?

BARRY LEIBA: Yeah, I'll defend Google. The problem is that the issue is with such a small fraction of the email that they get, that it's not that they don't care, but they have other priorities that they need to deal with. What I really got up to say was just clarifying one thing on the dbound stuff. I think there were three proposals at some point, and there were two main ones, and I think there was a third one. The main thing there is that there are so many different use cases for this list, and the different proposals optimize for different use cases, and settling on what the working group really needed to do was never resolved, and I wish we could get back to it, because as you said, I think this is a really important piece.

JOHN LEVINE: Yeah, but we weren't very good at defining the problem, which turns out to be something the IETF isn't very good at.

BARRY LEIBA: So, that's sort of my way of saying maybe some people from ICANN can help us get there.

JOHN LEVINE: Yeah, people who actually are down in the trenches with like making the certificates work and making the cookies work would be really great. Well, I seem to have stunned everyone, I seem to have stunned almost everyone.

BARRY LEIBA: Stunned, or worn out, right.

JOHN LEVINE: I am here all week, and I'm pretty easy to recognize, if you have questions later.

EBERHARD LISSE: Rod Rasmussen, there you go.

ROD RASMUSSEN: Rod Rasmussen, SSAC Chair. I want to plug SSAC. We did SAC-70, the Board actually accepted SAC-70 and said go do this, which is great, an IANA registry that would kind of supplant or support Mozilla; nothing happened.

JOHN LEVINE: Having talked to some of the people running the PSL, it turns out to be a much harder problem than I had understood, particularly because much of the software they use as the PSL is unbelievably fragile, like add one space and browsers will crash.

WES HARDAKER: Wes Hardaker, USC-ISI. Thanks for the presentation. As much as I keep staring at this stuff all the time, it still fails to stay in my head permanently, because of the complexity and how much you have to add, and I certainly miss the days when I ran an SMTP server on my laptop and was able to deliver mail straight from my laptop and watch the queue much more easily than my remote distant server. Has there been any published statistics about how much all of this stuff has actually helped? I know Google and lots of them fight it, but it would certainly be nice to say this is the percentage of spam we marked beforehand versus after and false positive and false negative rates, and actually a decent study. I haven't seen one, but I suspect there has to be one.

JOHN LEVINE: It would be really hard to do for two reasons; one is that every large provider's spam filtering is a deep, dark secret, and the other is the open heart problem, which is like at the same time we invented SPF, we invented DKIM, the spammers were changing their techniques. So if we had some academics that wanted to take a whack at it, and we had providers who are willing to do anonymized data, then I think it would be a really great topic to do, but I would not underestimate the difficulty of what you would be attempting.

WES HARDAKER: Accurate labeling would be one of the tricky aspects of it.

JOHN LEVINE: Actually, the users help us do that. One of the big differences that webmail made is that now that the mail provider can actually see what you're opening, and they can see when you move stuff in and out of the spam folder, and they use that to tune their filters.

WES HARDAKER: But you are making the assumption that the user is accurately marking their stuff in and out of their spam filter versus deleting it. They're not great at that.

JOHN LEVINE: You're correct, but there is a statistically useful signal there, particularly at large volumes.

WES HARDAKER: Okay, thanks.

EBERHARD LISSE: Okay, one more question? And we're done. Thank you very much, I really appreciated this one. Jaap Akkerhuis is next.

JAAP AKKERHUIS: I'm Jaap Akkerhuis, I'll talk about compliance testing with internet.nl. I'm actually proxying a lot of people here. It's not really my project, but NLnet Lab is involved. Internet.nl's website, you can go to it if you want to. We have in Netherlands, we have the internet standards platform. The internet standards platform is a platform from a lot of different companies, internet society, for standardization and that's a government club which actively promotes the use of standards in various sectors. So this is a separate identity for the internet standards platform.

Basically, the idea about how to deal with internet standards is not make mistake about first do a law and then go and change everybody who violates the law, because it's just like a checkmark exercise, and perhaps nothing is really happening. The idea is that people should comply, whenever the government wants to get service or a new contract for existing service, the suppliers are to comply to the standards, or at least explain why they don't follow the standards. And so this is a way to get people to support the standards and really do something, instead of filling in checkmarks, and it seems to work pretty well. So, that's the background. Well, you can say comply or explain, but what does comply mean? How do we check? That's where the internet standards platform was born. It is a way to test out whether or not you comply to standards.

So, that's what it is. It's supposed to be a user friendly way to test adoption of standards. Compliance test is not really exclusively a security standard test, but it is what the government likes to see implemented. It's also to be used by end users to see whether their suppliers are complying to what the government really would like to see happening. So you can check your own bank and see what they do with the internet standards. It also helps to create huge demand for using these standards by creating the awareness. You can call to your ISP and say why don't you fill in the blank here?

There are three categories in which compliance is being tested. Just the basic connections, how to connect to the internet, and how the web server is behaving of some supplier, your bank, the government itself, or whatever, and also email standards, how the way you do email is adapted to the standards. There is also scoring, this scoring is relative. It is possible to have a 100% score, but it's not really required for everything, 100% is ideal, but 90 is good, as well. Because there might be service standards that you cannot fulfill the complete test, you've got an old system. Internet.nl also give you an explanation why some of the tests fail, if they do fail, with some suggestion of improvement.

But again, there might be a valid reason for failing. It is purposely not a debugging aid or a learning tool. You can do some debugging, but if you want to review serious debugging of how your system is set up, that's for the supplier to do. The end user really wants to know how the system behaves in total, and by that you might find some debugging stuff, but it's not its first purpose at all. A lot of people are asking all the time, can you do more debugging? There is a limit, but then it gets too complicated for the end user, that is why there is a limit.

Furthermore, the scoring is designed by a committee which runs this platform. People might have different interpretations how important things are and how much it will be weighed in the cost. And in the end, NLnet Labs we do the test, and we don’t really do the scoring, at least not the basis of the scoring. So if you find things wrong, you go to the committee and complain there.

Okay, how does it work? Well, as I said, it's a website. And apart from the usual information what you find there is three boxes where you put in either your web or your connection or your email and then you just click the button and the thing starts to run. So, what comes out of it is the scoring. So, here it is for registration.icann.org. And you see a score of 91%, and there are some things missing, and if you scroll down you can look at the basics. But note, this is just what the website does, how you contact your website. It is not really what the service is. Because if you look at ICANN.org in your browser you will notice that you have to be redirected to a different website, and that scoring is less, it only scores 51% of the total.

I show this to show you that even if you get 100% score, it doesn't really mean that the service is completely safe to use, because there might be a lot broken. For one thing, there is a problem here with IPV-6, and there is no DNSSEC, I believe, and there are some other things, not really great. So here are the details of some of the things. I would suggest that everybody really going to have a look at their own favorite website and see what the score is, and you will be amazed about how bad some of the score is for what you thought had always been fairly reliable. There are similar scores like this for email, checks things like DKIM and DMARC and gives hint how to make things better. According to John, there is something wrong with how the DMARC scoring is done, and he complained to the committee and we'll see what comes with that.

Anyway, what the government is doing this as a batch test. It's not really made for public consensus, but it does periodically test all the government domains. There are some plans to make this an open API so other people can do tests as well, but it will take a lot of work to make this being used for the public, and nobody really wants to pay for that, so it's not sure whether it's ever going to happen. But doing this for the government actually shows some interesting statistics. The adoption rate of the various standards. These are the standards and the adoption rates of the various parts of the email standards. You see, it is still going up.

The DMARC is the latest addition to what used to be for compliance. This is only for the government agencies, things like that, it's not for the rest of the Netherlands. They do this maintenance every six months just to see the metrics how they approach to improve security and reliability. So what we're noticing is that there is some competition between the various sectors of carrying the highest scores, which is kind of fun. This was planned to [inaudible] but it is actually already done, a complete redesign of the whole system so to aid translation. There used to be a Polish version of this site, but it's very difficult to maintain and the guy who did the Polish translation went to do something else, so it was hopelessly altered, very hard to do the translation.

Now it's believed to be much better, because other countries have found interest in doing the same stuff. It is completely outsourced now, it is released and it is available through Github, so everybody can get it. So, it's not a lot of work to make it easy for people to adopt. There are some set if installation instructions, but they are difficult to do. But what has been done is that there is a docker image, so if you want to do a quick and dirty setup, you can at least take an existing docker images and do that, and so you don’t have to go through all the steps of setting all the parts of the system. Anyway, have a look at Github. Some more details see the website itself, and if there are questions about the whole system, don’t ask us, we are just the back office for doing this stuff, but go to Internet.nl and they should be happy to find out, to do the questions for all this. I guess this is it. Questions?

EBERHARD LISSE: Thank you very much. Very nice. I like the idea that it not just shows IP but it also shows you what's wrong with my mail server. I just saw that it uses some outdated cypher tools. The Dutch government, it's referred to documentation issued by somebody in security in the Dutch government, which is in Dutch only, but when you click on the link you will find there is also English translation of the text, so this is very helpful and it is well written. I think this is another thing that will come handy in fixing my own situation.

UNKNOWN SPEAKER: I have poked around at this site too, and I feel that the tools are very useful. It wasn't too hard to tweak my web server to get all my websites to 100%, and then I tweaked my mail server to get it to 96% and the last 4% they were wrong, but at least in each one they tell you what they want you to do, so you can decide intelligently whether you want to make that change or not. So, even though it's not perfect, it really is very useful.

EBERHARD LISSE: We have a remote question?

UNKNOWN SPEAKER: We have a remote question from Dennis Tan. Are there plans to include email address internationalization standard to the test suite? If yes, can you give some rough timeline? If not, can you elaborate as to the decision not including it?

JAAP AKKERHUIS: Well, no international added, there is not a lot of, let me put it this way, in the Netherlands, the international language doesn't really take any discussion, so there is no drive from government to do anything on that. However, the requirement for what is proper international AIE is not really that clear, as well. These things cost a lot of work in tweaking email and money, and so if somebody really wants that we add this, as well, they probably should, maybe some money would help to do that. There is no reason why we should do that from the start. There is a limited amount of time we can spend on this. But, complain to the committee.

EBERHARD LISSE: Okay, thank you very much. Our next presentation is from Abdalmonem Galila. He likes to stand when he makes a presentation, which is perfectly order, so he will now tell us about IDN, EAI and Universal Acceptance.

ABDALMONEM GALILA: Yes, before I start my presentation for Universal Acceptance, first I would like to identify what is the difference between user acceptance and universal acceptance. All internet enabled application devices and systems should comply with user acceptance, but at the same time, current systems, current devices, current applications use APIs developed 20 years ago, and these APIs were not designed to comply with international domain names and even the new domain names. That is why we should step forward from user acceptance to universal acceptance. So, everything is about the end user, all of us agree about this. That is me, I am Abdalmonem Galila, I work as Deputy Manager of .Masr “مصر “.IDN ccTLD “NTRA” of Egypt.

My background is technical. Yeah, it is my right, I wanted to access my local content, my Arabic content website with my own local language, is Arabic, for example. I wanted to send and receive email in my own local language, it's my right. For example here, this is a domain name in Amharic, one of the most popular languages inside Africa, and of course I wanted to keep my identity as I am from Africa continent, I will use, for example, http://dnsforum.africa. So let us turn our look back to two months ago for the event of Transform Africa Summit 2019. Today 50% of the world is connected to the internet, but it has taken 50 years to get to the level. Are we going to take another 50 years to connect with the rest?

So, my question for you now, do you think that non-Latin languages are among these barriers, 50 years to be connected online? So, the contents for my presentation today will be about four main topics. The first topic will be about Internationalized Domain Names, what is the idea behind this concept, and why Internationalized Domain Names. What is the perception of internationalized domain names. For email address internationalization, what is your thinking also about this? What is the idea behind that? And of course, I will talk about the new top level domain names. Now we have top level domain names related to city characters, we have .London, .Istanbul, .Africa, so as I said before, the ABIs used by the current system developed 20 years ago, is it able to handle all of this?

And of course there is a lot of issues with the current system. Who to care about this issue? There is a community initiative supported by ICANN called Universal Acceptance Steering Group, I will talk a little bit about UASG, and I will talk about the concept of universal acceptance and how to make your application as a service provider ready to handle the internet user, how to make your application ready to handle IDNs, to handle EIs, and even to hand new top level domain names in English. So there will a conclusion and we will open the floor for question and answer.

This pie chart approximately represents that English content exists online. If you think a little bit about why there is need for a multilingual website. One main reason for this, if you ask Amazon about this, they say our sales is increased 100% if we add more languages to our website. So as answer for this exercise, currently it is 50% English versus 50% non-English. But before it was 75% English versus 25% non-English. And there is an expectation that in the future it will be 25% English versus 75% non-English. Take a red line here and go to the next slide. I will go through some statistics. English is not the only one for the top languages used online. We have Chinese, Spanish, Arabic, Portuguese, and a lot of other languages. Do you know how many languages inside Africa? More than 2000 languages inside Africa? And there are four countries inside Africa have half the population of Africa, and these four countries don't use English, most of them don't use English, so we lose business.

Next slide, we only have 26% of the worldwide internet users are English. Let's look at internet users growth from 2000 to 2018. The user growth for English is only 3%. For other languages, for example Arabic, 36%, for Chinese it is 10%, Japanese 14%. So, let's go for Twitter. Distribution of twitter usage in the Middle East and North Africa in 2016, by language, more than 70% of the tweets using Arabic language. The user of Twitter advanced more than users of Facebook. So they use the English one, but they prefer to use their own local language. As I said today in the morning, when you are going to ATM machine and this ATM machine has more than one language, has more than English, you have English and your mother language, which language would you prefer? I would prefer my own local language. I will go to Arabic, of course, as I trust Arabic. So, what are the messages driven from the above exercise?

Let’s go back to before the current situation, it was 75% English versus 25% non-English, they would like to use Google.com for example, this email address and other domain names you will use English, but it will differ in the future, when it will be 75% non-English versus 25% English. They will use the domain name in their own local language. By the way, this domain name is in Arabic. It is the domain name using .Africa. So, that is the idea behind internationalized domain names. That is why internationalized domain names is important. There is the idea behind we are thinking about internationalized domain names. So, the internet has evolved. The landscape of top level domain names has changed markedly. Since 2006 we have more than 1300 new generic top level domain names. These are not just in English, you have non-English letters, not just two or three characters, root limited to .com, .net, which are ASCII characters, we have .istanbul, .london, .africa, more than three characters, and of course the TLDs is not static, you can add TLD frequently, and of course the mailbox itself, no longer to be ASCII, you can have mailbox with your own local language.

For example, this is my mailbox in Arabic, not just using ASCII letters. For example, .asia, .com in Arabic, .amex, this other language, I don't know what is it, by the way. So, why we should have international domain names? So, let's go for the larger provider of international domain names, they said that using the

IDNs will help the customers to remember a domain name more easily, will help to better communicate and market these verbally, and are easier for local customer to use. Will help better reach the local customer base. How does usage of international domain names? Domain names should be displayed correctly in a familiar way, IDN domain names accepted when I want to have a hosting space. ASCII domain names and IDN names should be treated equally.

Search engines, social media applications, software tools, browsers, should be able to handle IDNs correctly. So let's go for email addresses. We know how important is email. How much Email do we use daily? There are more than 200 billion emails sent and received per day worldwide. More than 8 billion emails sent/received per hour worldwide. There is an expectation to be 5.5 billion worldwide email accounts expected by the end of this year. So, again, as I said before, we only have 26% of the internet users are only English. What are the messages driven from the worldwide statistics? Email is the most popular application worldwide, number of email accounts is still increasing, non-English users more than English users, that is the main idea behind why we should have email address internalization. I wanted to keep my online identity. I want my local language in Arabic with my own local language domain name and send and receive emails with my own local language.

What does the user expect from using email address internationalization?

For email client MUA as we said, should able to process EAI emails, I mean by processing, send and receive. Email servers should be able to process EAI emails. EAI address usage across the internet is the same as the legacy, the English address for social media, mail hosting, et cetera. Anti-spam engines should be able to do its work with emails in EAI, in other languages other than English. So, what are the issues with the current systems that used ABI developed 20 years ago?

So, let's go for one of the social media applications. I wanted to sign up for new account using my new generic top level domain name for my email address. It will not be accepted. Also, it is English. Let's go for EAI address, I want to start up using my EAI address, it will not be accepted. That is the issue we face now for IDN, for EAI, and for new generic top level domain names. How come you want to promote IDN, you want to promote EAI, you want to promote new generic top level domain even in English, and your service provider application is not able to handle these new generations.

So there is a community initiative supported by ICANN called Universal Acceptance Steering Group, founded in February 2015. The main goal behind UASG to take care about the issues related to IDN, EAI and new generic top level domain names. UASG is comprised of more than 120 companies like Afilias, Apple, CNNIC, GoDaddy, Google, Microsoft, governments and community groups. So what is the benefits of adoption of universal acceptance? There is more than 9.8 billion dollars annual opportunity as a gain of adopting universal acceptance. This amount of money came from two parts. The current user who cannot speak English well, they will use and want to keep their identity, they will go to the new generic top level domain names and the other user who can't write English, he will go for EAI. So, universal acceptance ensures that all domain names and email addresses can be used by all internet enabled applications, devices, and systems.

So the question to you now, if your system for opening business tools for the billion internet users. So, what are the target systems. You have to ask yourself the question now, UASG is scared about their own application? No. So have to ask yourself, does my application have a domain name or email address? Does my application read or write domain name or email from a file? Does my application read or write send and receive domain name or email address from online service? All these applications if you answer yes, this means UASG is caring about this application. So, how to make your application to be universal acceptance ready and open your business to the next billion internet users. There are five criteria. The first criteria, accept, second validate store, process display. Accept, you accept the domain name or email address through online service. For example of Facebook, you will try to validate top level domain names, it is more than two or three characters, it will be rejected. So the extension should be updated to handle that TLD not just two or three characters. So after you make validation for the data you have for the user, you should store it. Maybe you're not going to store this inside the database, so this database should be Unicode enabled. So your application, your webpage should be Unicode enabled.

So the summary, Accept, validate, store, process, display. If you have any issue inside your application going to ask something, you need to know how make your application ready, you can submit ticket to our website here at uasg.tech. But there are a lot of resources here available to make your application ready. For example, maybe you are with hosting provider and you wanted to be sure that the domain name is already working, it is already there. So you can get the root zone from this link. You should of course follow the protocol IDNA 2008, not go back to 2003. I want to talk a little bit more about the difference between IDNA 2008 and 2003. For Unicode of course you should identify which letter is accepted by your registry, by your application. You should go to unicord.org to see which code is allowed.

This is my appreciation, this is Don Hollander who was General Secretary of UASG. All UASG members around the world appreciate this man after his retirement from the work for UASG and he is now enjoying his own life. You have my appreciation, Don.

EBERHARD LISSE: Thank you very much. That is what I would call a par force tour through a complicated topic. Any questions? The question is of course, should businesses be responsive to their clients, and in the end if there is a demand will there be a supply? Alright, next will be Jiankang Yao from .cn who has spoken here before, many years ago, and who will talk about EAI deployment in China.

JIANKANG YAO: Hello everybody, my name is Jiankang Yao from cnnic.com. I would like to share some EAI deployment and suggestion. Abdalmonem gave a very good presentation about why should there be EAI. So in cnnic.com we have done a lot of contribution in efforts to make EAI deployment in China. So I would like to share some information. First, EAI Promotion from CNNIC. EAI standards were published in 2012. After the standards we invite

Coremail, the top email service provider in China to implement EAI. We have CNNIC events together with other regions to have EAI testing. So, our CTTV News, first Chinese email address was sent by CNNIC. We also got APEC project, APEC is multiculture registry for different languages. We use APEC funding in 2014, we have EAI meeting in Beijing. We invite Google because Google in 2014 implement EAI in Gmail. CNNIC provided further EAI testing with Coremail. In our conference we have some for EAI testing. We are also co-testing with Microsoft with significant contribution to this area. We have EAI accounts as a Gift from Professor Qian from Chinese Academic Science. Also an EAI email sent by Robert E. Kahn, so I'm happy to participate in this historical event. We also have Chinese National People's Congress representative called for EAI support. We also have some technical seminar in Peking University, one of the top universities in China. I gave presentation about EAI, introduced the idea and knowledge. Classmates is from computer science department, some are postgraduate students. To my surprise they don't know EAI. Because our computer books or test books they only know .com, .net, but don't know .china, or a lot of new GTLDs they don't know. We also invite them to some testing, for example give them name at Chinese email address. Young students say it is very cool, they would like to use it. But before this seminar they don't know anything IDN. So in the future, education and educational material is very important.

So we also have for promotion IGF EAI meeting in 2016. We invite Microsoft and USG and ETDA from Thailand and will cohost. This will enable every user with a unique Internet Culture ID, so will promote EAI and IDN. We also hold Suzhou EAI meeting also we invite some from Germany, India, and Thailand to join this discussion. These are some pictures. Last December we promote Chinese domain name application. So local government also gave very good support. We also support and encourage open source MTA to adopt EAI standards. So, current email service provider situation, current Xgen there is a lot of news about XgenPlus from India, so there is news XgenPlus beats Gmail, Office and others to bring Unicode support in email to India. XgenPlus in India has allowed to support EAI, they gave every civil servant in India opportunity to use EAI address for government business. So, Yandex from Russia also supports EAI, so also CNNIC also provide EAI platform with Coremail. For international email service provider, CNNIC has already got progress. For example, Gmail in 2014. Also we have hotmail last year, now if you have address you can send and receive from hotmail. Also Exchange email server already support EAI. Some news, icloud and Yahoo mail in the near future will support EAI. So the biggest email service providers in the world will support EAI soon.

So I think there is also some problems. The problem, some email service providers, they don't support EAI account registration, 700 million users, their interface still do not support EAI. In future we need to make email service provider to support EAI. For example my name at Chinese email address to send and receive from other email service provider's email address. There are three phases to support EAI. First is according to UASG definition, Abdalmonem just gave an introduction about this. Phase One: Can Deal With EAI Message. For example, accept, validate, store, process. Currently Gmail, hotmail, support EAI in Phase 1. They can send receive message but they cannot provide EAI client or EAI account. Very, very difficult to provide EAI account. Current they have internal systems use email address as ID, so current ID system is only ASCII based. They need to upgrade to support Unicode. But in future, step by step, will move to support.

Phase 2 Can Register with EAI accounts. Internally for open free email service providers. Big email service provider never support this kind of function, only small some small service providers.

Phase 3, EAI accounts as an Internet ID. For example in China our very famous payment system, AliPay, or AliBaba's payment systems. They use email address. We encourage them to double as Chinese email address to enter as account. So suggestion push major email service providers to enter Phase 1. Also we like to push some open source to have a platform to provide EAI. In the future I would like to see ecosystem of open source because the user wants to service their email system, and they can follow the guidelines step by step. So I think mostly important is to learn from seminar. Education is very, very important. Domain names now can be Unicode so in future we maybe have some education, so submit some education material for EAI is very, very important. So if some experts have created some books for university students to learn what is EAI and IDN. So thank you for your kind attention!

EBERHARD LISSE: Thank you very much. I think what I'm getting from this is that if such a big organization CCNIC puts some resources into it, some outreach and PR, politicians is always good. If you put some elbow grease to it, it actually will start penetration. I particularly like the thing with the students. It's probably a very good thing to go to the universities and teach the younglings this is what's available and they will start making sense of it. Any questions? Alright, thank you very much. And finally, we have got Tim April, no you're not presenting, no, you're not Tim April. Tim will present the paper he did this morning already. He stood in for the people from Mauritania who didn't tell us they wouldn't application and then during they tell me they would appear again, and I'm not letting them present today.

TIM APRIL: I'm Tim April, I normally work for Akamai. I'm also involved in a couple other projects, and I'm a member of the SSAC, as well. And I'm here today to talk to you about the DNS Transparency Project. This is a project, if you were here this morning, this is all just a repeat of it. But this is a project that spun out after some recent high profile attacks on the DNS that happened over the last year, and trying to make a system that can help us notice these sorts of attacks in the future. The mission of this project is to create a system that makes changes to the DNS visible to end users and anyone who is interested in any updates to specific names in an audible way.

So right now, DNS, if you want to monitor, it a pull based system where you have to go and run a dig or some other process that's sending a DNS query and receiving response. Then anything that doesn't fall within your monitoring window may be missed in that case. So if you're pushing a change to a TLD that updates every 5 seconds or 10 seconds, if they push a change, wait 20 minutes, and push again, and you're only monitoring every half hour, you will likely miss that change in the zone. So right now the registration flow for any name on the internet would be the registrant talks to the registrar or the registrar reseller, which then pushes ultimately up to the registry and then into the name servers that serve that zone.

Resolvers will then query that zone where any change from the registrants submit the change to the registrar to when it's pushed to the name servers, any change that happens along that path that's not what the registrar requested, could result in malicious behavior in that zone. That would then be picked up and cached by any other resolvers for however long your TTL is. As I was mentioning, with moderating tools, if you were to build a monitoring system for your zone you probably pulled in a number of different places, either you used some sort of API to pull data out of your registrar. If you had a relationship with a registry you might pull that, or you could actually just pull the TLD name servers and see what responses you're getting back. That's mostly to bypass the resolver caching that may give you data that is old.

Some systems also do monitor the resolver just to make sure you're getting response back. And then that monitoring tool will send you some sort of either an email or some other realtime notification hopefully that change has happened. Whether that change is good or not is up to the registrant to determine. So the solution we're working towards now is to create an open system that will receive data from the registries and possibly even registrars and push notifications to the end users that something has changed.

So this is the diagram of what we're proposing to build right now where possibly the registrar and then hopefully as many of the registries as we can get access to will push data into what we are calling the DNS Transparency Project, which will then process that data and replicate to whomever subscribes to it. That could be the registrant receiving just an alert of changes, it could be companies like MSSPs, managed security service providers, where they will pull in data related to their clients and then send them notifications otherwise. And that's kind of the model where we're thinking someone would consume the full feed of data rather than just filtered data.

The solution is based loosely on the certification transparency model, which is what the web PKI has been moving towards for many years, where whenever a web server owner requests a certificate from a certificate authority, the CA will now push information about that certificate into a transparency log that is maintained by a couple different organizations. The problem with CT, certificate transparency, that prohibits many people from using it in a helpful way, is that there is no tooling that the average end user or small business can install to monitor their domains of interest, and that's where part of the tooling that we're looking at building stands in, where if you're a small to medium business you can go just subscribe to the domains you're interested in and get updates on those. Monitoring CT takes some significant effort if you want to do every CT log that exists.

So, as I was saying the input we're looking for is data from registries, so any zone updates we are specifically not looking for contact information, we don’t want to get into the mess that is GDPR and we want to stay as far away from that as possible. We're also looking for the data to come in as near realtime as possible, so that we can notify users as quickly as we see the change so that they can be informed about what's going on. These are the outputs we're considering, we're currently working towards trying to get a raw feed of domain changes so essentially a publish subscription service where we will publish all the zone changes as they come in, fragmented by RSSAC likely by domain 2, so that you can see if you're example.com you will get an update every time example.com comes in with the current state, the last state, and we'll compute the difference and include that in the data that comes out of it, as well.

And then there is also the filtered updates where it's just domains you're interested. So you can subscribe to your own domains and then also if you're relaying on other critical domains on the internet, like if you're using a CDN, you can subscribe to the domains that are involved in your chain into the CDN system. And then we'll plan to try and send those alerts through multiple different messaging systems, because if you send it through email and you're email system just got compromised, who knows where that message just went. So right now the current status of this project is we're working on creating an independent entity, so a nonprofit in the United States is the current plan that will hold all the process and development and the relationships with the different data providers, and then also manage the terms of service for using the system.

And we're also starting to build the proof of concept system in collaboration with a few different registry partners who we have tentative agreements with to actually start doing this data sharing. If you're interested in helping out, if you're a registry, I'd love to talk to you this week, if you're interested in helping share data. If you're a registrant we're going to try and post information to the webpage that we're in the process of building right now, it's not fully stood up because we're still very early in this process, but if you're interested in subscribing to the proof of concept, let us know.

And then if you're a company that's interested in subscribing to the data feed or helping out in any way, let us know. Down the road we're going to be trying to set up a self sustaining organization that may need money or in kind donations, and things like that. So, if you're interested in learning more about it, we're going to try and get that website actually useful in the next few days, right now I think it's just a blank page, or you can email the information at dnstransparency.org and that goes to a bunch of us that will be happy to answer questions or talk more. I think that's all I have.

EBERHARD LISSE: Well, I am appreciative of a fellow user of graph whiz to write your presentation. I may have misheard or not heard, why are you doing this?

TIM APRIL: At the end of last year and the beginning of this year, there were a number of sophisticated attacks on the DNS that were hijacking domain registrations for very short windows of time that were missed by a number of different monitoring systems. So it got a bunch of people, that we've been working together on this for thinking about how to make DNS changes more observable in a way that can be acted on and understood. So essentially an early warning system that something is going on. Because these attacks were unnoticed for many weeks.

EBERHARD LISSE: But the answer to those attacks were the DNSSEC, as I understood it. It was only sites that were not secured properly that were hijacked.

TIM APRIL: I believe there were some changes, there was nothing technically prohibiting that attacker from changing the DS records in that zone. The DNSSEC benefit was coincidental and lucky, it wasn't a perfect solution.

EBERHARD LISSE: Alright, thank you very much. Any other questions? Thank you very much in particular for stepping in on such short notice. And now Andre Filip will give us a little wrap up, we have done this all the time, but since Kobe, we have decided to steal his notes and write a proper report of it. A short one will form part of the ccNSO after actual report and the proper report will be attached to the presentations on the website.

ONDREJ FILIP: Hello everybody, I'm the last one who speaks here, I'll try to be brief. Let me wrap up a little bit what we have learned today, because there were many interesting ideas. We started with a presentation of Mario from Mario at registro.it and he brought a hot topic because startup is not implemented in all of the registries worldwide, there are still some that don't know how to tackle this issue, so he presented their way how to deal with it, and he presented four interesting parts of the RDAP chain, the validator, query, server, and client. So I think it was very useful for many of you to encourage you to start this service.

He also had a very interesting idea and that's the client altered configuration based on the server. I think that will be pretty challenging to do, but that was an interesting idea. Then we had a presentation from Brantly Morgan about topic which is also emerging. We are from kind of the old internet and we don't use cryptography so much as those guys who do everything based on blockchain. So, he presented one of the cryptocurrencies called Ethereum which has its own Ethereum name service and he presented the idea how to link this stuff with old DNS so it was pretty interesting, and they passed the test with .xyz so they're on their way and maybe you can be inspired and you can cooperate with those guys.

After that, Jacques Latour who presents quite often, so you know him very well. He presented a document from SSAC-105 which touches the kind of relationship between IOT and DNS in the security area. He also described whatever bad can happen with the small things that are not considered to be computers, but in fact they are, so there is a lot of risk related to that. And he was a little bit frustrated that not so many people read the document, so if you can please do it, he will be happy.

Then we saw a presentation from Patrick Jones and we saw what ICANN is doing to spread the knowledge about DNS about DNSSEC. So he presented the training that they do for regulators, decision makers and businesses. So not for the people that are very often here. So they are traveling around the world and doing a lot of activities. And then last before break was from Jay Paudyal who kind of explained, introduced us the problem of IDN for many of us who speak English or languages based on Latin, it's quite to understand it, but it's a very important topic. And as he pointed first, then many others repeated after, the English language is declining on the internet so it's a pretty important topic to help the others to use their own natural language.

After the break we had a pretty long and very useful kind of introduction, or I even I would call it tutorial from John Levine about email security and the DNS. And it was perfect so we learned many techniques how to kind of prevent spam. It was very useful, whatever you can do to stop these things that annoys all of us. So, thank you very much John it was really interesting. Then there was a presentation from Jaap Akkerhuis about a very interesting and helpful tool called internet.nl. I hope you all checked your websites and you have some notes from that. Then we had another two presentations which are related to internationalization, first was from Abdalmonem Galila from Egypt and it was not just about IDN, but it was more importantly about the EAI. So again, it was pointed out that English is declining and how important is to support those internationalized email addresses. So interesting presentation.

That continued later on with Jiankang Yao from CNNIC and again we saw how many problem he has, but he also pointed out what are the successes, what they did helping this out, how they supported open source MTA implementation, some providers, and so on. Also we saw three phases how to support EAI. So it was very inspiring what we can all do to help this, and I was taking notes myself and thinking of all services CA provides, whether we are compliant with all those issues. And last but not least was the presentation from Tim April about DNS Transparency Project. Again that might be some interesting projects that may help many of us fight with DNS abuse. And

Tim presented how this will continue and how this will evolve. So that was all. I have to thank everybody for coming and I need to especially to thank our Chair, Eberhard, who created the program, so thank you very much Eberhard. And that's all, have a nice evening.

[END OF TRANSCRIPTION]