ABU DHABI – How It Works: Understanding DNS Abuse
Monday, October 30, 2017 – 10:30 to 12:00 GST
ICANN60 | Abu Dhabi, United Arab Emirates

STEVE CONTE: -- questions, and I hope there's new questions coming out. If you didn't attend yesterday, then this is going to be all new and exciting to you guys, and I appreciate you coming and showing up. Today we have Dave Piscitello from our office in CTO department, the SSR division, and he's going to go over some of the DNS Abuse work that we're looking at and that we're working on.

How we run it today; if you have any questions, we've got two mics. I'll be watching, if you have a question, we are doing this with remote participation, so please raise your hand, and I'll bring a mic over to you and you can interact with Dave. We do encourage it, so please help us make this a dialog and not a presentation. It's meant to be a tutorial, meant to be a workshop and we're looking forward to your input and your questions as well. So with that, I'll turn it over to Dave.

DAVE PISCITELLO: Good morning. My name is Dave Piscitello; I'm the Vice President of security and ICT coordination at ICANN. I also, as part of my job, am part of the 24/7 Security Operations

Community, so I spend a fair amount of time working on DNS Abuse with people who are from the private sector; large companies, multi-globals who investigates phishing spam, global botnets, and I also work with people from law enforcement, from various departments of justice, because we all form a community that we call the "trust-based collaborative community" that works very hard to try to keep many of these threats from affecting your daily internet activities and also to keep you from harm from those threats that are most particularly targeting you for financial harm or for fraud or for some other criminal or miscreant behavior.

So Steve said that this is a repeat of yesterday's course; I have probably never given the same presentation twice in my life so I'm sure that there'll be something different from today if you do want to stick around. And of course repetition is the key to understanding, so it's always good to come back and get a refresher.

We're going to cover basically four different topics today; I'm going to talk to you a little about what DNS Abuse means, and then in the ICANN context the way that we distinguish abuse from misuse. I'm going to give you some examples, they are semi-technical; we're not going to go into the actual operations

of the protocol, but I'm going to show you how the attacks are conducted, sort of a chronology or an anatomy.

Then I'm going to talk about abuse in the ICANN context, and that specifically is going to focus on trying to understand what you might be interested in attending during this week at ICANN so that you can learn a little bit more about how does abuse factor into policy and governance.

And then, the last thing I'm going to do is kind of give you three examples of current attacks and show you how the criminals are using the DNS, how we've used the DNS to actually go back and defeat the criminals, to kind of give you an understanding of the counter-play that exists and how critical DNS is to both legitimate actors for pursuing their internet wants, and criminals for trying to exploit the fact that we want things from the internet to their, typically, financial advantage.

Please, if you have a question, feel free to raise your hand, we do need to use the mics as Steve said, so if you can be patient we'll get you a mic as quickly as possible; Steve is very, very fast.

So answering the first question; what is DNS Abuse is actually a little bit tricky because it's hard to actually find a globally accepted definition. In the ICANN community, we typically talk about malicious conduct, and that's because we try to embrace

all the different ways that people might misuse the DNS or abuse the DNS in a fashion that would either lend harm, or cause disruption to what is an important mission-critical infrastructure.

Some people call DNS Abuse just “hacking,” some people call DNS Abuse “cybercrime.” There are lots of abuses that are broader than cybercrime and I’m going to show you some of them. And the other challenge that we’ll talk about is, what exactly is cybercrime?

The threats that we’re going to look at against the DNS, if you’re a security person, you usually draw a risk and threat matrix, fall into the areas of data corruption, denial of service, and privacy violation, and so we’re going to look at some examples of those.

DNS misuse is something that we actually distinguish at ICANN community to try to distinguish people who have made an intentional action to try to register a domain or use a domain that they acquire through a normal e-merchant process, and from the people who actually have a legitimate registration and domain name and it’s exploited by someone else. So in some cases we’re talking about the intention to deceive, we’re talking about the intention to misuse or to cause the DNS to do something that the original registrant didn’t intend.

To keep it a little bit simpler, DNS Abuse for us is attacks or abuses against DNS infrastructure. DNS misuse is exploiting the protocol or registrations for some ulterior motive. So why is the DNS a target for attacks? Can you make money off the DNS? Well, you're not going to get any money out of the DNS but you can get money out of people who are using the DNS by impersonation. You can also get money by coercion.

So for example; how many people type in an IP address 3, 4, 5, 100 times a day to go to a website? Nobody. When I first started in 1983, yeah, that's what we had; you knew all 127 sites that you actually wanted to visit. So everybody types in a domain name, we don't even do that anymore; we use Siri, we use search, we use buttons that we push on our phones to get to an app. So what's going on in the DNS is actually relatively not visible to us as much as it used to be, but it's still the way that we identify and locate content or mail recipients, or mail delivery destinations on the internet.

So if it's that valuable to people, it's that valuable to criminals because this is something that a criminal can exploit and he'll exploit it in a couple of different ways. One is if the criminal actor or the malicious actor intends to do some sort of harm and disruption for notoriety or for hacktivism, he will try to disrupt the service, because if he causes the DNS not to work, most of

our little app buttons aren't going to work and if you ask Siri a question she'll come back with an answer that's basically not the one that you want to hear. Or Alexa; I'm not biased on any product by the way.

If you can exploit the DNS, if you can make the DNS lie on your behalf as a criminal, you can direct people who intent to go to eBay towards a criminal site, and this is a typical phishing attack on Vector, or you can lure somebody to an impersonation of a bank login page and get into their account and transfer their money. So tricking or defrauding or deceiving users is very, very important to criminals when they consider how they're going to use the DNS as one of their tools in their tool kit.

The vectors for exploitation are: malicious registration of domain names; for example spammers go and register many, many, many names for illegal pharmaceutical campaigns, and these names will look like "smarthealingstore.tld,""meds24x77cheap.tld," and on and on and on, and certain illegal pharmaceutical gangs so to speak will register hundreds of these names.

And in fact, you can find them very easily if you go into your Google search or whatever other search you want, and type "no prescription" and choose a .tld. If you choose .edu, you'll probably find about 68,000 to 98,000 responses almost every

single day because the criminals are actually going and hacking into WordPress sites at universities because they're vulnerable, and then when they get that WordPress site they will create a page that says, "Online pharmacies, click here." If you click there, it will take you to what's called an affiliate page.

The affiliate page is usually one of those names; "Smart Healing Store," "Meds 24/7," and these are pages where you can go and you can purchase prescription drugs without a prescription. Often they're not their actual drug, often the amount of medication that's in those drugs is not the exact prescribed amount, the stabilizers are not necessarily created in a sterile environment, so there's a lot of danger in that whole marketplace and that's one of the reasons why especially people who are investigators from the United States, like me, have an obsession with trying to eliminate those people.

The other way that you can have a vector for exploitation is as a criminal, I could try to hijack someone else's name resolution, and if I hijack someone else's name resolution I can redirect a DNS query away from the intended site to an impersonation site, and I'll show some examples of how to do that.

And then the last is to just literally corrupt the DNS data; take over a host, like a name server, and put information in that was not intended. Or poison the cache, and we'll talk about that.

Some of you might have been to the introduction, the DNS session yesterday, but just quickly so that we have a context for today let me talk about the three operational elements of the DNS that are typically targeted by criminal actors. The first is what's called the "client or stub resolver"; this is software that's in every device that connects to the internet that uses DNS. So your mobile phones, your laptops, large servers, every device that you can use to connect to the internet that resolves names to numbers has a piece of software called the client or stub resolver.

Sometimes it's in the operation system; Window and Linux for example have a DNS stub resolver. Some browser manufacturers like Google have a specific resolver for their Chrome browser, and so there are lots of these little resolvers and they are essentially the piece of software that asks questions to the DNS. And the questions typically are for users, what is the IP address that's associated with this domain name?

So, if you're going to eBay and you type www.ebay.com in your browser, your browser may formulate a DNS query that says, "Find me the DNS resource record that provides me with the IP address that corresponds to www.ebay.com."

Who do they ask questions to? The DNS is a distributed database; there are tens of thousands if not hundreds of

thousands of authoritative name servers, probably millions actually. There are millions of recursive name servers, so who do you ask questions to?

When your device connects to a network, for example when you power up your laptop and you connect to the WiFi network here, you receive information from what's called the "dynamic host configuration protocol server," a DHCP server. Part of that information is typically the name or address of a recursive name server. Recursive name server now becomes your slave; he is the one who runs around the global DNS infrastructure and finds the answers to the questions that you ask using the DNS protocol.

So clients ask questions, resolvers answer questions. Authoritatives are where the data reside that people want to access, it's called "zone data," and each registered domain name has its own authoritative, or authority, and that authority generates the information that associates names with addresses for the authority.

So eBay for example has an authoritative name server. In the eBay name server there's a record that says, "If somebody wants to go to www.ebay.com use this IP address." And when the DNS queries, the way it works is called "recursion"; the client asks the recursive server, "What's the address of www.ebay.com?" The

recursive actually goes out and starts at the root and says, "Can you help me, because you're the only one I know?" He then comes back and says, "Well, I can't tell you anything about www.ebay.com, but I can tell you where the com servers are." He goes to the com servers next, the com servers say, "I can't tell you anything about eBay, but I can tell you where the eBay authoritative name servers are." When he goes to the eBay authoritative name servers, they say, "Well yeah, sure, www.ebay.com is at this IP address." That's returned to the recursive, and that's how the DNS works.

So you now know that this is a very, very distributed, very, very fast system, but there are lots of different places where criminals can actually go and try to disrupt or exploit.

So we talked about these three different kinds of resources in the DNS and these are the kinds of attacks that one can use against each of these elements. I can try to prevent DNS traffic from being delivered to any of those devices. I can block responses. I can do a denial of service attack against an authoritative or a resolver. In many cases, lots of you have home networks, you have a little router that you connect to your broadband carrier; attacking your broadband carrier and trying to make that unusable is another way to stop traffic from getting to you. Now, most people don't attack individual's broadband

networks, except gamers who get mad about some other 12 year old that beat them, but it happens quite a bit more on attacks of higher level of an infrastructure.

For example; attacking a government site by taking down their firewall or their routing, would be an attack that would be very common, or one that we try to make certain is not common, quite honestly. And when you are attacking one of those devices, you can attack it at very different levels; you can attack the hardware, you can attack the operating systems software, and then the same is true for name servers. I can attack the hardware, try to flood it with information, I can attack the OS software to try to force an error that was un-accommodated for in the programming. The same thing is true for name server software.

In those devices that have what's called a "local storage" or a "cache" I can attack those as well. And then the same is true for any application that I might use.

The last two are very, very important because one of the things that is most challenging in an internet of billions of users and almost the same number of devices, is configuring them all to be safe, configuring them all to operate all the time, is nearly impossible so there's always some element of software development or administration or configuration that one can

exploit. Any questions about the difference between DNS abuse and misuse and some of the attack vectors?

So let's talk about some examples, and what I'm going to do is I'm going to walk you through a couple of these typical attacks against name servers, or recursive name servers. They fall into a couple of categories. One of the things you can do, as I mentioned earlier, is you can try to deny service by targeting a particular computer like a name server, and cause it to fail, much like those of you who remember Windows blue screens of death, or a MAC OS pinwheels until you have to interrupt or do something, you can do the same thing functionally by attacking hardware that run name servers.

You can also do what's called "exploit to own," and in this kind of attack the criminal or attacker is looking to take over, take control of a particular piece of the infrastructure. You're all familiar with denial of service attacks, right? So denial of service attacks; there are so many. I have a long three hour version of this seminar; I spend a whole hour on the different kinds of denial of service attacks. There are elements of the denial of service attack that I'll talk about in a moment, they're called "reflection and amplification," and the combination of those two enable what's called the "distributed reflection and

amplification attack" or a "DDOS attack". So we'll see how that actually works.

You can also just try to exhaust the resources of a particular host, or you can try to cause a cache not to work efficiently or cause a cache to provide the wrong information; we'll see some of those.

So it turns out that it's almost trivial in some cases to cause a DNS name server that hasn't been properly patched to fail, and that's because the people who write software are not infallible, they can't write perfectly secure code, and the people who are attackers invest a very, very large amount of time trying to discover an error that they may have made. So we call these "bugs" or we more formally call them "vulnerabilities", and when a vulnerability is used to cause harm, or to cause a disruption, we call that an exploit.

So there is a database called the common vulnerability and exploit details that's available that has a full list of discovered vulnerabilities and the patch that's available to them once we've discovered it and we've recovered from that error, because the software manufacturers go and try to find a fix to the software. Unfortunately, not every software manufacturer creates patches right away, and remember I mentioned configuration and administration? There are still people who haven't patched

software for which a vulnerability was known three, four, five years ago. There are still people who are running software on the internet that provides name resolution that is three, four, five years out of date.

So somewhere in the internet it's very likely that we can go and find CVE 2002, 0-400 and use that to attack a name server and cause it to fail. And what this does is it sends a DNS message that's not correctly formed; some of the bits are changed, the purpose of changing those bits is to make the software at the server react in a way that it didn't expect, so it causes an unexpected condition at the name server and the name server responds by shutting down, because the logic was going, "Oh, when I get here I faint." That's basically the way it works, so this is what's known as an "exploit to fail attack".

An "exploit to own attack" is very similar; except the purpose is to cause the software at the name server to be put in a condition where what it does is it offers a remote administration opportunity. And you might have heard these as "remote execution", "arbitrary remote execution attacks" if you follow the literature. What the malformed message or stream of messages does is it creates an opportunity for this attacker to literally become the administrator of this device, so that's an "exploit to own".

Why would you want to own someone else's DNS name server? Anyone? To change it; exactly. What would you change? You could change the records to do something that you wanted to do. So if I could break into a government name server, I could go and I could change the address of that government name server's web address to a defacement site. And so someone is trying to go to some government, would try to go to the web, and instead be redirected to a face that says, "Hi, this is anonymous, we defaced your site, your security is horrible, and by the way, we don't like these political policies."

So that's a very big embarrassment for any government, and so trying to keep your name servers up to date and patched and running the correct software and properly secured is extremely important. Any questions about how exploit denial service attacks work?

The distributed reflection and amplification attack builds on two kinds of attacks that were individually run years ago, and what happens normally is that an attacker begins running software. By the way, you can actually download denial service software all over the internet; if you want to become a DDOS client, it's very, very easy, but what attackers normally do is not wait to recruit people, but they recruit computers by infecting them. We'll talk about that in a moment.

Once an attacker has decided on a target, he finds the address to that target, and in this case it's the address of a DNS name server, 10-001; he then spoofs as his source address the target's address. So he's essentially lying, and every one of those computers is saying, "I'm him, send all the responses to me." The spoofing takes place because we have very, very poor admission control in the internet; internet service providers could implement what's called ingress traffic filtering to prevent addresses that don't belong from entering the network at locations they don't belong.

So the attackers send the DNS query, which is relatively a small message; it's usually under 100 bytes, and the answers aren't going to come back to the attackers, the answers are going to go over here. So if I can do this at gigabyte rates, I can send across my thousands of attackers, I can send a gigabyte worth of traffic of answers. And the answers are normally small; they're normally a couple of hundred bytes.

So if I really want to exhaust bandwidth, what I do is I try to use a name server that perhaps I've deployed that has a particular kind of record that I know is very big. So in the DNS there's a record type called TXT; it's very easy for a criminal to take one of his compromised computers, make it an open resolver, and then put a TXT record of 4000 bytes. So now I can send thousands of

small queries that will result in thousands upon thousands of large responses being directed towards this targeted host.

And what's going to happen? As soon as the traffic exceeds either the access bandwidth or the capacity of the network or the capacity of the name server, the name server is going to start struggling. And then he's going to have to start dropping traffic, but his performance is going to lag, and this is what a denial of service is intended to do; it's intended to disrupt or slow traffic.

The other problem that this name server has, if he's dropping traffic, he can't really distinguish very easily in many cases whether he's dropping the bad guy's traffic, or he's dropping legitimate queries, so responses to people who are actually trying to get name resolution from here legitimately are going to be dropped as well, so that's another aspect of the attack. Any questions about this?

UNKNOWN SPEAKER: I'm just wondering if DNSSEC is going to prevent this attack or still not guarantee?

DAVE PISCITELLO: So, DNSSEC is not designed to defend against this kind of attack. This attack it is very, very challenging to defend against. One of

the best defenses would quite honestly be to have all the network traffic that comes into the network only come from legitimate sources; from addresses that are assigned to the access providers. Unfortunately, that requires compliance across all the networks in the world, and that hasn't happened today.

Another way to defeat this is to not allow what's called "open recursion". An open resolver accepts queries from everyone. If you don't accept queries from everyone, then an attacker whose network spans the globe and comes from thousands of networks can't use that; he's got to use one that is more easily identified, so really the way to stop a denial of service attack is to not allow people to be anonymous in their traffic behavior, and to be able to localize the origins of the traffic and apply backpressure.

And all the good denial service mechanisms try to get around the fact that these constraints don't exist, and they do things like "back scatter analysis", which means they go and they try to find out where the majority of the traffic is coming. To actually mitigate a DDOS attack, you need to contact your upstream provider, your upstream provider needs to try to start working with the other providers who are seeing the same traffic, and

you have to slowly kind of push the traffic back. Operationally it's really, really costly.

My personal opinion is that all these operators who are reluctant to put in ingress filtering, or it's also called "source address validation", are actually causing harm and receiving harm, because if they're the targets of a DDOS attack, they're just as much victim of other people not doing source address validation.

This is still a big problem; there was an effort a few years ago to try to reduce the volume of open resolvers, and I think we dropped it down to like eleven million from twenty-one million. Well, eleven million is still like way too many by eleven million.

Another way to do that same kind of reflection attack is to use what's called a "resource depletion attack", and this uses a different underlying protocol to DNS. Normally, the DNS runs UDP, which is a datagram protocol that doesn't have any state. A UDP message just goes and gets delivered; there's no confirmation of delivery. TCP has a delivery confirmation and reliability element to it. So that means that the party that's sending, and the party that's receiving have to come up with an agreement on what this connection is going to look like.

So to begin this kind of attack, you open a TCP connection with a packet called a SYN. The SYN packet is basically a request, "Please, let's create a connection." The targeted host will receive the SYN, and he responds with something called a SYN ACK, which says, "Yes, that's really good, I'll do that." Now that's fine, except that the targeted host doesn't know yet whether the originating host received the SYN ACK package, so in TCP we have what's called a "three-way handshake"; what's normally the case after the SYN comes back, is that the originator of the connection sends a SYN ACK himself. And so that completes the three-way handshake; both sides know they can hear each other, and they can start sending traffic.

When you do the spoofing, and if the SYN ACKS go out, the traffic gets dropped because it's not going to a listener. If the attackers all withhold the SYN ACK that's part of the three-way handshake, this machine sits there waiting for that third message. And when he opens up his connection, he reserves a certain amount of memory in what's called a TCP connection block

So attackers said, "You know what? If we can send like thousands and thousands of these TCP connections and make him reserve memory, way beyond what he really anticipated, maybe we can just force him into a state where he doesn't have any memory left." So this is a memory exhaustion attack. The

attackers send as many packets as possible asking for connections, this guy reserves memory for the connections, and eventually the memory gets exhausted.

If the destination is actually smart enough to say, "I'm going to stop at a certain number of connections," the attacker's purpose is still satisfied because the name server that's on the receiving end can't distinguish legitimate people who are trying to establish a TCP connection from the criminal, and he's going to be dropping connections. And eventually most of the connections, typically the case, most of the connections that end up being completed are attacker connections. And so the amount of legitimate traffic to illegitimate traffic is small.

This is supposed to be an animation, but we're using PDF so I'm going to just walk through this; poisoning a cache is actually a very old style attack, it's been around for at least 20 years. And one of the ways that we see it today is that an attacker may launch its spam campaign. And so you receive an e-mail message, and the e-mail message says, "New diet pill, guaranteed to lose 15 pounds in 27 minutes." And somebody out there is going to say, "Wow, that's great!" And they click on the link.

When you click on the link, "loseweightfastnow.com", your browser, or your mail is going to launch a browser, your browser

is going to ask for the IPv4 address of "loseweightfastnow.com" so that comes through my local resolver, goes out to the internet, but it's going to go to a name server that a criminal is using. He's going to go into that zone file for "loseweightnow" and he's going to say, "Okay, I'm going to send him back this IP address, 192.168.1.1."

There are three pieces of a DNS response message, one of them is called "additional section", and the additional section is as the name would suggest, a place where you can put additional information that might be helpful to the recipient. Most of the time, it's other bits of information about the name that you're querying.

So if I were querying "loseweightfastnow.com" a legitimate use of the additional section would be to send back the names and the IP addresses of the name server, or the mail server. However, the criminal says, "You know, maybe I can go and I can cause him to do something that I would benefit from." And he says, "By the way, in the additional section I'm going to tell you that the address of www.ebay.com is 192.168.1.2," which is a lie.

But, when this message gets back to the local resolver, if the local resolver is caching, he is going to say, "I'm going to take the response that I got, and this guys was kind enough to give me an update to my record for www.ebay.com, so I'm going to put that

in my local resolver; in my cache." So now, the attacker has achieved two purposes; one, he's got somebody to come to his loseweightnow site, and he could be hosting malware there, or he could be selling crappy drugs. But the other thing he's done is that everybody subsequent to that query who comes and asks to resolve www.ebay.com is going to go to this address, which is probably the attackers address, and is probably hosting an eBay phishing page.

This is a known attack; people have known it for years. It's easily circumvented by configuration, and it's still perpetrated. Questions about this?

This is an attack by the way, sir, that actually DNSSEC does defend against, because this criminal name server cannot correctly cryptographically sign this record. The caching resolver will try to decrypt the signature, and he will fail, and he'll say, "I'm not going to accept that record because it is cryptographically incorrect." So that's one of the attacks that DNSSEC does defeat.

You can also poison caches that exist on individual computers. There are two ways to do this; remember I mentioned DSCP and how your device gets the address of a name server, along with its own address when it connects to a network? If you click on malware, on a malware link or on a malware attachment and

that malware happened to be something called “DNS changer”, which was used about four years ago, DNS changer would actually go into your configuration and modify your name server; your local resolver. So he’s literally redirected you right at your host level.

The other thing that DNS changer evolved to do was go into a file on most Windows computers called “The Host File”, and it would add entries to that host file. And the way that Windows and Linux have worked in the past is that before going to the DNS, if there was a local entry, you would go and you would read this host file and use that file. And so what would the attacker put into the local host file. He would put IP addresses corresponding to names that he wanted to intercept and direct elsewhere, so banks, or eBay, or PayPal; and so once he’s got control of this, the attacker can actually go and sell parts of his network to different attackers for different purposes. And so this was the beginning of using spam and botnets as a criminal infrastructure, or part of the evolution actually.

If the attacker is more of a social engineer than a protocol engineer, he may try to do something a little bit more soft than actually trying to contrive ways to break into machines, or to break machines, or to break software. So domain name registration hijacking is one of those kinds of attacks.

Do any of you have a domain name? Has anybody registered a domain name? Just a handful. Alright, let me walk through the typical process for a gTLD. And I'll use myself as an example; I wanted the domain name securityskeptic.com, which is where I host my blog, and so I went to a registrar. When you go to a registrar online there's usually a page that has a web form to check whether the string is available, and the registrar asks the top level domain, so in this case the registrar said, "Is securityskeptic in the com domain available?"

That request is issued out using a protocol called EPP, extensible provisioning protocol, the com said, "Yes, it's available," the registrar told me it was available, and I decided to register it. So I paid a fee to have that name for one year, and then the consequence of this whole merchant transaction is that a billing record was created, a WHOIS record containing the registration information was created, a DNS send to was created to create an authority for me. It identified a name server that was going to host my zone data so that I could put in www.securityskeptic.com is at my IP address. And then there was some other information.

Now, what are the characteristics of this for most commercially transacted domain names today in the gTLD space? The process is highly automated because you want to do order fulfillment

very quickly, it's rapidly provisioned, you can often get your domain name up and operational and have a web in place in less than an hour. All the correspondence that registrars actually use is e-mail, so this should be warning signals to us all. And then, there's also an opportunity for criminal actors to register volumes because registrations are relatively inexpensive, and they're also using stolen credit cards that they've obtained on the dark web. But it seems that even with that they still go low price.

So this is actually extremely good for consumers; we can grow the market, there are lots of people who have websites. But it's also good for attackers. Attackers will go and register their own domains; they're inexpensive, they're easy to do, there's not a lot of validation, it's relatively easy to patent false information into WHOIS because the obligation is on the registrant to provide that information.

So attackers will, especially spammers, register domain names in volumes, sometimes hundreds, sometimes thousands, and they're going to use them for all sorts of purposes. Phishing or fraud pages, ransomware payment web pages, malware distribution sites; so these are sites that basically you visit because you were lured there by an e-mail message that had a

URL, when you go to that URL it's hosting a free anti-virus software in a zip file.

So you say, "Oh cool, new anti-virus software, best on the market," and you've got all this great advertising and your 80 year old Grandma only knows that you told her to go get anti-virus. So she goes, and she pulls down this anti-virus.zip file, opens up, and ha-ha; here's the malware or the ransomware and Grandma's now typing in her credit card to pay for the ransomware which is not going to have the effect that she anticipates.

Scam sites; advanced fee frauds, reshipping frauds, everybody know about these? You've got, "Hi, I'm the Lord Premier of the Presidio of some country you've never heard of, but I have 31 million Euros that I need to transfer out of this country into the United States. All I need you to do is send me 500 Euros for the attorney fees." Okay, that's rather sad.

Counterfeit goods sites, we're all familiar with those, and illegal pharmaceutical or piracy sites. Those are the direct attacks; these are the names that they register for all those kinds of attacks that people can see if they were looking. And if you were paying a lot of attention, you probably wouldn't visit. But there is a lot of deception in those emails; some of them actually come

as URL shorteners on text messages, there's lots of ways that criminals hide or obfuscate their actual intent.

The other things that domain names are extremely important for for criminals are building global spam infrastructures, cloud infrastructures for spam delivery, and botnets. And so they need name servers for those; that's one of the purposes of having names, they need names for their commanding control because if they put IP addresses into the malware that is delivered, those people in my community who are experts in reverse engineering, the malware can see the IP address rather quickly. So they all include some sort of algorithmic mechanism today that generates domain names at a very large volume every day, and then the criminal goes and registers a handful of those. Any questions?

Hijacking is actually, you know, why pay if you can crack or if you can socially engineer? One of the things that you can do is impersonate a registrar as an attacker. And you generate a phishing e-mail, and often the attackers use anticipated or a regular correspondence from a registrar like the WHOIS accuracy update, and say, "Your WHOIS is inaccurate, please come and update it or you're going to lose your domain."

So the social engineering that they're playing on is fear of losing your domain, and people will say, "Oh my God, I don't want to

lose my domain," and they go and they hit the URL, they visit an impersonation site of the registrar, and then they enter their credentials, and the person updates his WHOIS and he's happy. Meanwhile, there's a man in the middle who is the attacker, who is sitting there saying, "Okay, I'm going to capture your credentials and now I'm going to use your domain and I'm going to change your zone file to support my criminal activity."

So that's a phishing attack; sometimes there have been instances where people who are real soft attackers socially engineer the help desk at a registrar reseller to get the credentials. That should never happen, but it does. Never underestimate the power of persuasion of someone who knows how to deal with human beings and manipulate them psychologically. And if you forget that as someone who's doing investigations, you are overlooking one of the first and obvious steps to pursue. So that's what the attacker will do; he'll publish data.

The other way that this is becoming more prominent is data breaches. One of the things that criminals can do today because of the volume and volume of e-mail accounts and other breach accounts like Equifax, is they have user names and passwords; how many people actually re-use the user name which is often an e-mail, and how many people use the same password? So

just going and taking your four hundred million of these accounts and going out and trying them everywhere; you're going to have a certain amount of hits.

So this is just to show you that each of these data breaches has possible other consequential actions, not just here, not just in a bank, not just in your healthcare privacy records, and it's one of the reasons why people are so concerned about trying to build in data protection and build in sufficient incentives for those who host things like password databases to be very, very careful about how they protect that data.

Another technique that criminals use to evade detection is something called Fast Flux. And what the criminal will do, once he's built up his botnet, his army of infected computers, he will host a web proxy or a name server at a given address or set of addresses. He'll set what's called a-time-to-live field in the DNS, which identifies how long an address record is deemed to be valid, very short, like five minutes. And then after five minutes expires through automation of his own zone, he'll go and he'll move the content, or enable the content at a different IP address. And so, he's literally just moving around the block, staying one step ahead of the investigators as he's playing this game. Often, this is done with web proxy, but sometimes it's done, as well, with DNS and the web proxy, and that's called double flux.

The last thing we'll talk about -- how are we doing on time? The last thing I'll talk about is DNS as a covert channel. One of the things that you can do, as a criminal, is host a name server, an authoritative, and you can write software for that name server to, essentially, accept labels as content. And so, let's say, I host Example.com, and I'm the criminal. I'm going to use Example.com to exfiltrate information. And what I'm going to do is I'm going to infect computers that I think have databases, and I'm going to try to pull data out of those databases.

So one of the queries that may come from this setup is Dave.Piscitello.ThreeMyrtleBankLane.HiltonHead.SouthCarolina.ZipCode.Example.com. Please find me the A record. It's going to go to Example.com and the nameserver, which is a criminal, operated name server, is going to simply extract all that information because he understands that this a record of personal data from a database.

Now, that's kind of obvious. People are going to start looking at their DNS and they're going to say, “Hmm, that's rather strange. So why are we doing this?” And you would catch that. So criminals actually either use what's called Base64 or encryption to hide the fact that these are personal data records. So that's one way to actually extract information from a SQL database.

You can also use that same technique, by the way, to run something like Telnet or Secure Shell on top of the DNS protocol. And people use that to bypass WiFi paywalls. So if you are only going to try to Telnet to your remote machine, which is very low bandwidth, or run SSH to your remote machine from a kiosk or a Starbucks, where they're going to charge you money, DNS is almost always running. So you can always just run DNS, and if you put a little bit of traffic, even one byte at a time, you can get a couple hundred bits per second worth of traffic out of DNS and that's enough to do human interaction through command line.

And so, there are many, many different software and executables that you can get on GitHub and other sites that will help you do this. Last year, around this meeting time, I wrote an article at the ICANN blog called, "What is a Covert Channel?", and it actually gives you some of these examples.

Malware writers can also use the DNS to pull new malware from a command and control or a DNS server to an individual computer. And what they do is use that record I mentioned earlier, the text record; they encode a chunk of an executable software in a text record, and then the malware that's on your infected computer goes up and asks for the text records. And so, he pulls down the text records, and then composes a new piece of executable software by chucking them together and then he

executes that, and that's how the malware gets updated, or refreshed, or told to do new things.

And one of the reasons why he may do that is because Port 53 is generally always open and often it's not inspected. This changing many, many commercial firewalls and many, many DNS firewalls now look at exactly what kind of traffic is going in either direction and look for anomalies. If you want to read a little bit more about how this works, you can go and look, just search on Feederbot or Morto.

Any questions about those examples? Are you all really depressed now? So we'll have to have some sort of rejuvenating energy drink to get you encouraged. Now that you know how bad it is, the plea from people like me is, “come help.”

So how do you help? Well, in the ICANN context, DNS Abuse is a very, very hot topic. We're talking here about many, many different aspects of abuse. The deliberations are not going to be nice and calm and just the kind of breakfast conversation you would have with grandma on Thanksgiving Day. They are challenging. People have business interests and commercial interests, people have operational interests. Technology people have questions about the sanity of trying to do some of the solutions that others think are appropriate at a policy level. The policy people see things in a very, very different way. And then

attorneys see things in every single way that makes them money. So it gets really complicated.

Topics within the heatmap of ICANN that are going to be close to red this week are WHOIS Accuracy, the General Data Protection Regulations and how that's going to affect the WHOIS. There's a public safety community that is going to be talking at the Cross Constituency Working Group about the needs of law enforcement and investigators to actually effectively combat all that horrible crime that I just showed you. And there's also several projects that are going to talk about reporting abuse or measuring abuse and how to use that information.

A large focus of what DNS Abuse ICANN community should be paying most attention to stemmed from the Government Advisory Committee and their recommendations and communiques to the ICANN Board of Directors. And essentially, on two occasions, the Government Advisory Committee said, "We really do need the registries and register operators to start paying attention to abuse. The ones that we find most concern right now are botnets, malware hosting, phishing, and spam."

I mention the Public Safety Working Group, this is primarily a working group that advises the Government Advisory Committee. It is largely composed of law enforcement and invited security subject matter experts. They put together their

own proposals for policy initiatives that cover GDPR, WHOIS Accuracy, Fast Flux, and DNS Abuse.

We won't go into too much detail on Carrier Grade NAT. This is hard to explain this if you don't have a lot of background in what Network Address Translation is and IP addressing is. But essentially, today when we use home access routers or access points, every computer that connects to that access point and then connects to the public internet, gets what's called a private address. And that private address is mapped onto one public address. So there's a many to one that saves the consumption of public IPv4 addresses, which are nearly exhausted.

What the carriers are now doing to save their own space is taking that private concept and they're assigning private addresses to their customers. And then inside one of their equipment, called a Carrier Grade NAT router, or switch, they're doing that transformation from private to public. The challenge for law enforcement and investigators is that now the public point of access has disappeared. We don't really know where this traffic is coming from. What we know is that it's coming from a carrier and then you have to go through court order or some other mechanism to the carrier and say, “Please give us your log so we can figure out where this traffic is coming from.”

So this a really, really big challenge. Carrier Grade NAT is an alternative to evolving towards IPv version 6. So the ICANN community and the address supporting organizations are also equally challenged in trying to encourage people to not take these temporary or perhaps short-horizon business useful strategies, but go for the long-term and use IPv6.

One of the things you may hear is how GDPR and WHOIS Accuracy and some of the other initiatives to help mitigate DNS Abuse will affect ICANN contracts with their contract parties, the registries and the registrars. There is already some specifications of abuse, or specifications in those contracts, to cover abuse. I'm not going to go into these in great detail, but here are the places where you can actually go and read if you want to know about them.

Here are some sessions that are probably worthwhile for you to attend. There's a Cross-Community Session on effective DNS Abuse Mitigation. That's this afternoon and it's probably going to be a very lively session. So bring Kevlar and whatever other defensive equipment you might have.

There's also going to be a talk about a recent abuse study that was performed by an academic group called SIDN. And their SADAG study, the Statistical Analysis of DNS Abuse in gTLDs, will be discussed. It was actually really, really well done. I was able

to participate and oversee that activity for quite some time. Since I've been working on a project that's similar to that in ICANN called the Domain Abuse Activity Reporting System, I can tell you that a lot of what they see and report is exactly what we see and report. So it's very interesting that this was done for the consumer confidence and trust obligation that ICANN has, and I think it's worthwhile going and listening to that presentation.

Then, there will also be a GAC Public Safety Working Group presentation to the GAC Plenary. I don't know whether that's public. Steve, is that public?

STEVE CONTE: Sorry, I just noticed, and I should have caught this yesterday, when Dave says March, he actually means October. Probably a modification from the other slide. The first one, I was just checking that the time is at 1:30 tomorrow, but I don't have the other one. So please, these are sessions going on, but the time and location and date might be slightly off.

DAVE PISCITELLO: This is John's slide.

STEVE CONTE: Ah, there you go. So please check the schedule. The schedule is authoritative, he's recursive.

DAVE PISCITELLO: See, poison cache. John Crane poisoned my cache. As you walk out today tweet, "John Crane poisoned Dave's cache."

Okay, the last thing I'd like to cover is three attacks that you may have heard of and you may be interested in them for three different kinds of reasons.

Obviously, my handle on Twitter is securityskeptic. I'm not cynical about things, I'm skeptical about things. And this is probably a more cynical statement than I ought to put up, but I've been doing counter-crime and investigating for over 20 years, it ain't getting any better. And it's sort of really, really frightening just how commercial and commodity we are beginning to see cybercrime. You can get Distributed Denial of Service attacks as a service, cloud offerings. Okay? Fast Flux and double flux showed up, probably, in 2000, 2008. They went away when we started figuring out how to defeat them. They came back with a vengeance on a scale that was two orders of magnitude larger and faster.

Spam is a cloud service. We're going to show you the avalanche of spam infrastructure. The internet of vulnerable things, the IoT

is just a train wreck and there's no stopping it, and I'll explain why. And I know that some of you may react negatively to me saying that, but one of the things that we have done through every succession of technology change, from mainframe to mini-computer, mini-computer to desktop, desktop to laptop, laptop to mobile, mobile to IoT, is we repeat the same mistakes. The people in each of these sectors don't learn from the previous developers, and I'll give you an example why.

And the last thing I want to talk about, and hopefully the slide actually will work, is the DNS as an ignition key or kill-switch for malware attacks. And I think some of you heard about WannaCry and WannaCrypt, and I'll try to show you exactly how that was fixed.

So honestly, not only can you go and purchase attack kits, or download attack kits, like LOIC or Slowloris, for your own fun or for your own vengeance if you're a gamer and you're mad, which is where a lot of these come.

By the way, do any of you play massively multiplayer games, like World of Warcraft and the like? So, one of the things that's sort of sad is that you have a whole bunch of people who form a group and they're part of a team, or they're part of a guild. And other guilds might be angry with them. And if they end up with some sort of confrontation in a face to face kind of battle, you

see people causing Denial of Service attacks against the guy who is probably the best player on the other team. And so, it's just really rather strange that what we've ended up with is an attack tool that is so ubiquitous and easy, that 12-year-olds can go and try to game the game by doing Denial of Service attacks.

But, other people can simply go to a booter or a stressor service and say, “I’m willing to pay $200 for you to attack this company for X number of hours,” and that's how it is. Now, the faster we put them on blacklists, and the faster we knock them down, the more they show up.

You can go to YouTube and you learn how to do this. I love that every one of these guys is an attorney, because they all say, "I am not responsible for any damage that you cause." So it's really fascinating just how this is growing.

Let me talk about Avalanche. Avalanche was a network that evolved from a major spam infrastructure, or spam campaign platform, that was botnet based to a malware delivery service. They essentially created an environment in a cloud infrastructure that was, basically, bulletproof. It was very hard for law enforcement to go and take it down because they kept moving. And so, one of the challenges was trying to identify the jurisdiction where they were actually hosting the infrastructure. And getting all the jurisdictions together was really massive.

It was predominantly used for financial fraud attacks and predominately ransomware. And literally, you could go to the Avalanche customer service site and you could go and get criminal domain registrations for yourself. You could get access to the command and control server that you would use to run your attack. And then you had your choice of 20 ransomware families. It's like, "Oh, today I'll use Teslacrypt." You know? And so, you're now on your way. You hit the engage button and you're now going out and you're infecting hundreds of computers, you're causing ransomware to be installed, the ransomware blocks people from using their computer until they pay a fee.

So the timeline for this Rockfish that I was mentioning earlier was that by 2008, investigators and commercial security vendors were effectively detecting and blocking Rockfish spam campaigns. So Rockfish kind of disappeared, Avalanche grew. Avalanche started in 2009. In three years it evolved to this malware delivery service that we were talking about.

For the next four years, law enforcement in 30 countries, private-sector investigators, ICANN, ICANN registries, 24 TLDs and 40 operators conspired together -- or collaborated, sorry -- collaborated to essentially identify a way to dismantle this botnet. By going through an exhausting number of mutual legal

assistance requests to obtain court orders for all these jurisdictions, for all the different sites and persons of interest, finally, in November of 2016, we had a simultaneous dismantling. That's a pretty long timeline.

Anybody want to venture why it took us so long? Jurisdiction, lack of common criminal law, just the natural slowness of mutual legal assistance.

Are there people unfamiliar with the mutual legal assistance concept? So it's essentially a way that a law enforcement agent in one country can ask for information or assistance from a law enforcement agent in another country. Often, when you ask this, the request will have to go up to your own Department of Justice or Crown Council or whatever. And then that Crown Council will go and ask through this formal treaty process for -- let's say the UK will ask the United States. When the request is received in the United States, someone in the United States has to actually go through and see whether or not the request is going to violate US criminal code in disclosing or assisting in this particular activity.

So that is not something that happens in real time. That is something that happens in months. Six months to nine months is typical. So now do the math and figure out how many we

actually had to do and you can understand that it took almost a year.

So actually, finding these infrastructures, identifying them, and getting all the intelligence that's necessary to execute this kind of dismantling can happen in much, much shorter terms. And so, one of the things that is useful for having more law enforcement engagement at ICANN is to try to understand how we actually cut the candle at both sides. How do we improve what ICANN community can do and how do we accelerate the process of actually getting the court orders, so that we can actually do these mitigations in much closer to real time?

I just wanted to give you -- this is an iCheck, but let me just tell you what this says. It says that criminal actors set up their networks in anticipation of attack and then they launch it. And often, people call that a zero hour, or zero day. In the first hours in onset of a botnet or a phishing attack, it's most often the case that private investigators are collecting the information intelligence and putting things like reputation block lists together to try to do triage, to prevent criminals from getting to user desktops, and user mobile devices.

Law enforcement usually gets involved when crimes are reported, when there's a financial loss, when somebody actually reports the issue to law enforcement. And sometimes, that's

investigators. Then the process of going out and actually converting all this intelligence to evidence, getting the evidence in front of the right jurisdictions through legal and court orders, or mutual legal agreements, and you can see where all the time is. And this can actually -- I will put a little bit of thought in your place. GDPR can change this timeline. And the reason why GDPR can change this timeline is because if private sector investigators need court orders to get information that they currently can get freely today, then we're way over here. And there's nothing that's going to happen in days or hours, it's all going to be months and years. So think about the balance that we have to strike to have data protection and not have other people harmed as a consequence of providing data protection to people who can exploit it.

This was the Avalanche outcome. Five arrests in four countries, 37 searches in seven countries. We seized all sorts of big equipment and we actually managed to grab 830,000 domains.

Yes, can you wait for a --

HEATHER COSTELLO: Thank you. Heather Costello here. I was just wondering what the right jurisdiction is to obtain a law enforcement? Is it the

jurisdiction of the attacker or is it the jurisdiction of the person who is the subject of the attack?

DAVE PISCITELLO: It's actually many jurisdictions, often. So, for example, where were the servers? Who were the registrars who were registering the domain names? Who are the registries who are registering the domain names? They have to be served court orders to go and block the name resolution of these 800,000 domains.

So this was 64 TLDs plus registrars. So, in this particular case, we're talking about a global attack. So there were people everywhere. There were four countries where we actually apprehended individuals. We, meaning the law enforcement and private sector. Thirty-seven searches in seven other countries. So if you just start adding what I can show here at the high-level, we're talking about nearly 100 court orders.

Now, in a single attack, let's say that it's just somebody who ran a phishing attack and it was hosted at a particular location and you knew the location, or there was child-exploitation material, that might be one-quarter in a local jurisdiction, go get it off the server, take the people into custody, and call it a day. Those actually are easy, and what GDPR wants to do, for example, makes sense.

These are the ones that make it all complicated. Because how do I gather all this information if I can't see it? And I'm not saying I'm not an advocate of privacy or anonymity. I'm just simply saying that understand the possible consequences of taking the privacy and anonymity and treating that separately in a regulation from accountability. Because if I register a domain name, in some respects it's like registering a weapon or registering a car; if those things are used in the perpetration of a crime, law enforcement generally takes action.

In the internet, it's slightly different because you have 168 countries that might be involved, all with different criminal law, but you also have thousands of private investigators who are feeding that information to them to help them accelerate that very, very long timeline that I pointed before.

SUSAN HERRING: Susan Herring here. While I'm sure there's no single bullet or single answer, but what's the ideal balance? Or what can be done? What do you think? To strike a balance between privacy but not without blocking law enforcement's efforts?

DAVE PISCITELLO: So there are solutions. I don't know which one is ideal. You will hear this week people talk about tiered-access. You will hear

people talk about vetted interveners. So one model that is used elsewhere for things like phishing is called the Accelerated Malicious Domain Suspension Program. And in that particular case, there are registries that sign up with the Anti-Phishing Working Group and those registries agree to accept attestations of phishing attacks from interveners that have been vetted by the Anti-Phishing Working Group. It's generally people who are very senior in security operations, who are very well-known already to the registrars, and it's almost like an expedited path to get the attention of someone who can take action on the name. That's one thing you could do.

Another thing you could do is create tiered-access. The challenge with tiered-access is that everybody wants a tier and everybody, eventually, wants the same information. So the challenge here is that, as I said earlier, you have business interest, you have operational interest, you have public interest, public safety interest, and you have data protection interests from governments, and they're all pulling -- and my personal opinion, and this is not necessarily ICANN's, is that they all pull away from each other and want what they want. And you can't make policy if everybody runs into their corner and says, “I want this, and I will accept nothing less.”

And my community of investigators is just as guilty of making that same mistake. And I'm simply saying, you have to put all three of these on the table and understand that if you do this, and you do this, the accountability becomes a challenge. So, before you go and do these two things, are you certain that you're willing to live with what results in accountability as a result of hiding this information in the manner that people are talking? And I don't know the answer. I've been struggling with this myself.

Of course, I live in the United States. Between the OPM breach and the Equifax breach and the fact that my state of South Carolina Internal Revenue Service was breached, all of my information is in the dark web. And so, I'm stuck. GDPR is too little, too late. It's going to be a long time before I'll feel comfortable that people can't steal my identity. And that's not putting a criticism on GDPR, but it's just the evolution of the way that we've turned the internet into, essentially, an arms race between criminals and legitimate actors.

Six minutes? Okay. All right. Let me -- I'm just going to do one minute on Mirai. Everybody remember Mirai? Mirai was this infection, or this attack against digital video devices and other devices that were directly connected to the internet and left a protocol called Telnet open to the public with default

passwords. So the lesson here was, why did you do that? This is something we knew in 1985 not to do, 32 years later you're doing something that we knew in 1985 not to do. Don't do that.

And the problem is that you have a whole new generation of developers who have never connected anything to an internet before because that's not what their business was. So there's got to be one of those little bars that says, "You must be this tall to connect to the internet, and you can't be down here. You have to actually have gone through this to understand." And trying to get new markets to do that is exceptionally hard, especially when you see the lucrative expectations of how many people are going to make trillions of dollars with their IoT offerings.

They're bad enough with just allowing access that allows somebody to take over the machine. Can you imagine all these devices, especially bio-devices, that are holding personal data? I mean, if they can't do this, do you really trust them to get cryptography right, to protect personal data on the device? So I just pound my head and go, "None of that's coming in my house." I have an old 1999 car that doesn't have anything digital in it. And we won't go through that. And I talked about that.

So WannaCry. WannaCry was -- unfortunately I hate -- yes.

HEATHER COSTELLO: So one of the things we're looking at doing within Australia is to come up with consumer trust marks that will, somehow, gauge the level of security that a range of IoT products might --

DAVE PISCITELLO: Right. So like Underwriters Laboratories in the United States?

HEATHER COSTELLO: Like, what? Sorry.

DAVE PISCITELLO: Underwriters Laboratories, United States. Your plug must be like this in order to be used in electrical system.

HEATHER COSTELLO: Well, it might be a series of stars. One star for no security, two stars for no more security -- anyway, up to maybe five. Something like an appliance energy rating or something like that. Do you think that could be useful? I think there's some models we can try.

DAVE PISCITELLO: I think with IoT, the danger is going to be that commodity pricing is going to make people make bad decisions. I'm in the wrong place to use the R word, so I won't use the R word -- regulatory. But I think this is really serious. If somebody is going to put something in my body, I don't want somebody who's never programmed in Linux to be the one who actually goes and takes the kernel and decides what applications go on my bio-device in my body. I want some oversight.

I honestly think -- and this is not speaking for ICANN -- but, my personal feeling is that this is where we're going to have a very, very important battleground between understanding how we allow innovation to move forward and whether it moves forward in the same relatively unrestricted manner. Because we're getting to the point where the possibility of human harm is raised much more dramatically than we ever have before. And I'll just leave it at that. We could have a conversation over coffee. And I don't want to destroy anybody's IoT market. That's not what I'm trying to do. I'm simply saying, you know, is this really what you want to put in your body? Is this really what you want to have you driving around and the like.

Go ahead. I think we'll just not do WannaCry. Let's go to the last slide.

UNKNOWN SPEAKER: This is similar to what happens in aviation. It takes many, many years to approve a software in aviation because you have to go through all these procedures. There's many times, do a thousand landings, there's everything to be very resistant to attackers and so on. But we don't see that in medicine today. They are starting to learn what aviation learned many years ago.

DAVE PISCITELLO: Right. And it's a good example for two reasons. One, it makes us all sober to the fact that, yeah, we all get on planes because we're pretty confident that they're going to come back down in one piece and we're going to walk off. But there are also people who are sitting there thinking, "You mean to tell me we're going to go through the same kind of hurdles for every little light that we want to plug into our house that we can remotely manage from our phones?"

So this is the whole spectrum of the conversation that's going to go on. But I think that certain sectors, like medical, or critical infrastructure devices are going to probably need something like that because you're also putting people in the position where a concerted terrorist attack on critical infrastructures can shut off the water, shut off -- you know, it's the whole post-apocalyptic nonsense that we see from the movies.

UNKNOWN SPEAKER: Hi, David, thank you for your speech. I'm [inaudible] from Taiwan. I'm also involved with the Avalanche operation. I'm just curious about how the Avalanche operation can work by blocking the malicious domain at the same time. It seems to mitigate the spread of malicious malware because not all countries joined this operation. Is the Avalanche coming back, such as changing the DGA algorithm?

DAVE PISCITELLO: Okay. So I think you asked three questions. But, let me see if I can answer them. One, so everyone can catch up, one of the things that some of those court orders did was they told the registry operators, "Here is a list of names that the criminals are going to register every single day, as long as the malware lives, and we want you to preemptively block them, people can't register them." And they're really stupid names. There's like 32 characters long, and they're all alphanumeric nonsense. So it's not something that's actually causing loss in the "Oh, I can't sell this to someone" context.

So all the registries that were involved that we knew complied. There were one or two that didn't. And I can't tell you publicly, and you know I can't tell you who they were, but ICANN was able

to actually do a bit of a facilitation because there were governments that didn't want to work with each other. And so, we said, “Well, we play in that sandbox with them, so we're going to get out of your sandbox, we're going to go over to this sandbox and talk to them and say, ‘it really would be nice if you would do this.’” And that worked. Now, that doesn't scale to a thousand registries, so that's a problem that we do have. So I think that's question one.

Question two is, what will prevent Avalanche from changing their algorithm? Nothing. It's going to be just another botnet. Avalanche is not the only botnet out there. There's, at last count, like 100,000 botnets. And so, there's probably anywhere from 50 to 100 million infected machines that are feeding some botnet or another. Possibly some machines are actually running multiple botnets. There are some people whose hygiene on their computers is so poor that they probably have every malware known to man. So you're not going to stop.

And this is one of the criticisms of doing what we do with these kinds of dismantlings, they are stop-gap. In the medical world, this is triage, this is like a MASH unit. We just go out there and we try to just keep everybody from bleeding completely to death. And that's not a good way to do this. I honestly think that we need countries to recognize that if we're going to stop

global cybercrime, we have to adopt a global common set of criminal laws. We have to agree on, not necessarily common penalties, but common processing, and we have to expedite the whole court order process. Because, don't forget, over the course of that four years, there were hundreds of millions of dollars, euros, and Dirhams that were extracted by fraud to all these people. So every day we don't act, it's costing people.

STEVE CONTE: So, we're at time plus two, so I'm going to allow one more question. And the slides that Dave has are already available on the schedule, so if you drill down into today in the schedule, you'll be able to get a copy of his slides. And Dave, would you be willing to hang out for a couple of minutes after this?

DAVE PISCITELLO: I actually have a 12 o'clock meeting, so I can't. Dave Piscitello ICANN.org, send me an email. I'm happy to try to have a conversation with you or share some email.

STEVE CONTE: So, we'll do this last question then, thank you.

UNKNOWN MALE: I only have a short question. Do you think that the information provided by the WHOIS service is the source of all the DNS Abuse? My question is relation with the debate on how to regulate with the WHOIS.

DAVE PISCITELLO: Okay. The question is, is DNS information the source of a lot of this abuse? That's a really, really hard question to answer. And it's also one that has like only 400 or 500 snakes between me and the answer. But, one answer is that there's a very, very big debate of whether or not people go to the WHOIS and grab email addresses and grab point of contact information for phishing.

I did studies years ago with a gentleman who just left, Steve Sheng at ICANN, and we could not find that. And I can tell you, anecdotally, that I have an email address that I just use for my domain names, I've never received any correspondence except from the registrar and from the mail service. So that's my own experience. Other people's experience seemed to have differed.

What is emerging, that is a concern, is that there's a technique called doxing, where investigators and even people who want to commit crimes are using social media handles and other ways to go and pull information out from as many sources as possible to then create a false persona and then engage with that person on

Facebook. So if I end up being able to know things about you, and I can go find your website, and I can see what you posted publicly on Twitter, and LinkedIn, and Facebook, I will then go create a Facebook account that looks like somebody that you might want to friend. And then I friend you and I will then go and try to figure out how I can eventually get really close to you, to the point where you share something you really didn't intend, and then I try to hold you for ransom or I try to use you to extract information from your company.

So there's a concern about that. Very, very legitimate concern. I don't know whether GDPR will actually fix that because people are just naturally going to share. And so, putting the obligation on the operator for what people share is a challenge.

But, that's some of the things that end up going in front of legislators and they don't quite see the connections, but what they want to do is stop it. And so, there's a lot of education left to try to get people to truly understand all these Rube Goldberg machine-things that if you pull this out, this goes this way, and you put this in, this goes that way and neither of the ways are the way you want to go.

Well, thank you all for staying. I hope you enjoyed the talk and I look forward to seeing you at another ICANN meeting.

STEVE CONTE: Thanks so much, Dave. So thank you all. The next session in this room will be at 1:30, we're doing a session on internet networking with Alain Durand. So if you have any interest in the IP addressing, the DNS routing, and how it all puts together, and how it all goes together, please join us back here at 1:30. Enjoy your lunch. Thank you.

[END OF TRANSCRIPTION]