TRP PDP WG – Small Group on AuthInfo Codes

Recordings from Wednesday, 14 July 2021 at 14:00 UTC
Audio:https://icann.zoom.us/rec/play/3fxe28avt9x2ZdvIvXQlC6r5svl0HM72HC5CHrn9kz6wp1pnw9eEHZxs-BCmHOuATJysKcfZsZeVTYB4.eeX3KNmCj-eIjYxo
Zoom: https://icann.zoom.us/rec/play/ysLk4ha7Qdo-HI9hvSsmeA9PkcMo1lF6cAzV5n_ZZZGbOVThsIYWYvJqpJHodRYipdyNwbwqVo22Z5iQ.gdF5KyvrhmoQBWhI
Zoom Chat (only recorded portion is captured in this link): https://icann.zoom.us/rec/sdownload/4Zloqou3HKhPygbuZkSMoTdWR0t2LkZb8c5NbVKhPF8GPbyeocI78_5vniqkKm6eR11JIHSCpDxUROg.-3TzcgBf29HP6jGd

Recordings from Friday, 16 July 2021 at 15:00 UTC
Audio: https://icann.zoom.us/rec/play/LNYuvqEBcHdtHC2BCJGFHNFGJzFvBJXzpSfUrbhfTY8SNUDtLD_KyuOmx6KDEuSEGVBW7B_pCuEyngo.cW-Ej_drLd2SSAG4
Zoom: https://icann.zoom.us/rec/play/S30UO_8jhp7-uypMHE6ZUBYvbmaMFP5KH0MEEuR-auUYncoUHDrrfFJyUJMFe4IgeC6QQprjl81yCuIt.DNru0WqQ89yrTtWE
Zoom Chat (only recorded portion is captured in this link): https://icann.zoom.us/rec/sdownload/2UZO7HKm8Q8597VwfH-XQclceqOi9mfW5MPRj9s_N-Ndhe4did7V8HTame819zTw3Zsd0BcrwPp653Sw.AadMj9pdmqHgwLw8

Recordings from Wednesday, 21 July 2021 at 14:00 UTC
Audio: https://icann.zoom.us/rec/play/TbLpB27A6oJqsOSJ55X8JlvLhcmYjt3y9dIW2k-RdRqcogm0XD3siXg9nISXYvTLAisaHvILyzvwkyxH.z2bYA0BTdhbUJBb_
Zoom: https://icann.zoom.us/rec/play/uVWjPVZunmKdgl5F-twssE-nDFPoh7i9-br6MY3vnNJ4M52kvfXgpQgCNiby4ORJZ8f6k6qKraiOYKZT.8WX9llj_Plf6PSdW
Zoom Chat (only recorded portion is captured in this link): https://icann.zoom.us/rec/sdownload/huJYZyHlmvyBsxR1yF5JPsW2yog9rfUJ7uUz4iS2p1pC1JOCg_kyCU16VghpgdMPLi9LyDTGpNh8hxAv.0zYW2uBOtAe05uOf

Recordings for Monday, 26 July 2021 at 14:00 UTC

Audio: https://icann.zoom.us/rec/play/oWDmptbuxiMXGdJHKBzYEPveASsg7d8Y76x0scsXQBVal7qcNWKYevPd7jnMG4aYh3pw6LTaTjmx9mQN.W40Rundm-4pyDKYY
Zoom: https://icann.zoom.us/rec/play/Cyq3TqsPkp_niYKhckhCsfqGK6sJr5SKAneYPzGyKxtNbz77pDXEVLT2eJ_879IynaWVYvfN6iohURJy.K7AxTBrdn8ZyFvKy
Zoom Chat (only recorded portion is captured in this link): https://icann.zoom.us/rec/sdownload/Cr3zT27p4_El_XTKenln2iSfwRK77_cBMTbxBO-ovRblAF70coX-JjnLu-dVryUMT7qsr0_Cb7agmAJQ.hoYihaOC6VWVvP2X

Notes from Wednesday, 14 July 2021 at 14:00 UTC

Goal of the Small Team: Trying to get to multiple scenarios of how we think AuthInfo codes should look in future, answering charter questions, and identifying any changes to policy.

Resources:
Working Document on AuthInfo Codes: https://docs.google.com/document/d/1O9PAnxWFUuPofLQCWIQXz8lT7KEj1HgH3b_obh0AK00/edit

See AUTH CODE SMALL TEAM BRAINSTORMING DOC:https://docs.google.com/document/d/1nwjgyvR3KNYqRojW_u4bEbGFemJBzM2NSlP7vNuhnmc/edit?usp=sharing
Brainstorming document includes responses to the Policy Status Report survey. This is the link to the survey results:https://nl.surveymonkey.com/results/SM-Q2J8JZRQV/

Discussion:
-- CPH TechOps – trying to make it easier or at least have some standards. Some of the questions around this might be to take a look at what we are scoping here.
-- Small Team to review the brainstorming document and add any missing context; consider answers to charter questions.
-- Question: What are we trying to do? Thought that we might ask TechOps to examine some use cases and provide some concrete suggestions. Then the TPR WG would look at that and translate that into policy requirements. But this sounds like the Small Team would prepare a use statement for the WG?
-- Answer: The goal is to come together on a solution/proposal that should be vetted with TechOps but is positioned so it fits well for them. That we've looked at the requirements and this is what we think we see as ask if there are holes.
-- Jim Galvin: Two things are of concern – 1) clearly indicated in the brainstorming doc: create more uniform and homogenous guidelines; example charter question b1, response of "no" does not supplant some principles/standards for uniform use and management of AuthInfo Codes; 2) example charter question b2, there is some discussion about the role of registries and registrars.
-- charter question b2 is probably the one that most needs to be discussed.
-- Jothan Frakes: For a different perspective, there are buckets that these questions could be filed under. Some attributes could be predictability, uniformity, security best practices, customer experience, etc. May want to come up with some **buckets** – like columns off to the right that have a tick box or not. There is some variations and standards would be helpful, but challenging to tell registries how to do something.
-- Perhaps those buckets you mention, Jothan, could be used to compare "beauty contest" contestants?
-- Jim Galvin: Two things 1) Word "control" – suggest to not use that term to talk about what we are going to do with AuthInfo. We don't want the registrar "police". Prefer roles and responsibilities. Decide what we want from the system and how we want it to work, and then what are the **roles and responsibilities**; 2) suggest for what we do here and how we do it: rather than being constrained by the charter questions, could we talk **about first principles**? What do we want most out of our solution? Example: How much do we want or not want a real time transfer system? Need a small list of first principles, think about the actual process of a transfer, then see if we've covered the charter questions in our new system – determine the challenges and whether we need fundamental changes.
-- Charter questions will answer themselves as we go through the process.
-- Premise is we'd use those as columns to the right of the survey responses and tick the boxes where applicable. This would provide some context to the various subsequent reviewers.
-- Suggest that the Small Team start by throwing out principles.
-- Jothan Frakes: TechOps is looking at introducing a high cadence of the meetings; there is a core group in the TechOps but is a different structure. On a parallel track with the work of the Small Team and the WG the TechOps can bolster its core group.

High-Level Principles:
-- Jim Galvin: What is the transfer process and what does it mean? Think about how we want to manage these steps: 1) registrant wants to transfer; 2) ask for a code from incumbent registrar; 3) give code to gaining registrar and say you want to transfer. The incumbent registrar has to generate

the code, give it to the registrant, confirms that the requester is the owner; and registry has to know the code. Gaining registrar has to initiate the transfer with the AuthInfo Code.
-- This is the way it works today.
-- Registrar has to initiate the transfer – registry has to be made aware that a transfer will take place; there has to be an interaction between the losing and gaining registrar. These are the high-level steps.
-- Maybe we don't have to make fundamental changes to today's system. What is important: if we like the system we have to add uniformity and security principles to the current system so all of the parts do what they are supposed to do, and that there is a compliance check.
-- We want to get a proposal to the full WG by the meeting on 27 July.
-- Jody Kolker: What happens when there is a fraudulent transfer? Do we need to be considering how to reverse a transfer quickly and easily and what kind of process do we set up for that? If we make this as easy and possible to transfer, how do we also make it easy to get the domain back to the rightful owner?
-- Roger Carney: Example doesn't have to add liability to the registry if the process is followed. Policy has to give them the authority to do it correctly.
-- Jim Galvin: Balancing a couple of things – in any kind of transfer you need a way to get it back. If you are going to move towards near real-time transfers, you have to make it really hard to move if it's not right to move. That's where you get into additional security. That means that a registrar who is the incumbent has to have procedures in place and security practices to ensure that you aren't transferring improperly. You can have claw-back process that is simple if you have secure procedures.
-- Roger Carney: You have to have a dispute mechanism for you to get back a domain from a fraudulent transfer. Are the multiple levels of dispute?
-- Jothan Frakes: If we are going to scout for things from the ccTLD world that are working we'll have to filter those through. Look at this through a holistic lense – if we are looking holistically across the process they do affect each other.
-- Roger Carney: This does tough on other topics, but we need to focus on AuthInfo Codes.
-- Volker Greimann: Inclined to leave a certain time window to abort/deny. Any process to recall a domain name needs to have a longer window. If you change to only the registrar can provide the registrar, you can't verify that the request is coming from the registrant. Should remove the registrant from the equation. Also how to handle AuthInfo Codes dependent on how to conduct bulk transfers. Could complicate bulk transfers.

ACTION ITEM: Small Team members to add principles to the brainstorming document, or add details and tie them to high-level
principles. See:https://docs.google.com/document/d/1nwjgyvR3KNYqRojW_u4bEbGFemJBzM2NSlP7vNuhnmc/edit?usp=sharing

Next Meeting(s):

-- Meet next week at the same time? Need more meetings? What can we accomplish between now and the next meeting.
-- Some are not available next week at the same time.

ACTION ITEM: Staff to schedule meetings for Friday, 16 July at 1500 UTC and Wednesday, 21 July at 1400 UTC.

Notes from Friday, 16 July 2021 at 15:00 UTC

- Following the small team's next discussion, we recorded minimum gross level steps and some buckets to consider.

- Jim provided some proposed principles.

- With these principles, tried to capture important security principles and important process level requirements.

- The first statement describes what the TAC is – an identity credential which represents authority to transfer the name.

- Next, there is a list of what is required of a TAC? When talking about security principles, was trying to get at what should be required of a TAC:
- Has to be unique per domain name and registrant – this is important
- Needs to be available upon request
    - MUST Have a TTL
    - MUST NOT exist until requested by registrant for a transfer – the important component here is that it cannot be guessable
- MUST NOT be stored by incumbent registrar – idea is it is created on demand
- Be submitted to registry in secure manner by incumbent registrar upon being provided to a registrant
    - MUST NOT be retrievable from registry – registrars can set a value for the TAC, but cannot go look at what the TAC was – this gets at the uniqueness principle and one-time use principle – always give the registrant a new TAC
    - Be removed from registry by incumbent registrar upon expiration of its TTL – there is not uniformity in what that TTL should be, but that's an issue for registrars to sort out – should it be a singular TTL or should it have different values?
- MAY be created by hashing the tuple ([SALT], [DATE-TIME], domain name) – registrar decides how to create this
    - [SALT] MUST be managed as follows:

        · Value must be a component of any TAC generation method

        · Access and use MUST be restricted to "need-to-know"

        · MUST be randomly created

        · MUST be changed regularly – if you create a SALT value (or random number), you have to ensure uniqueness – one way to do this is to hash values. You have to have a one-way hash with a random secret number, or there could be a cryptographic key that is changed regularly (or every time registrar generates the TAC)
- Have to disagree with a lot of these principles as this would break a lot of systems – for example – the principles of changed regularly and created when requested – this creates needless complications. This would introduce a lot of costs to implementation. Agree that it is a one-time use item, and the item is used when it is used for the transfer. After it is used, the registrar creates a new one. Do not need to change how auth codes are currently being handled.
- No matter what we do it's likely to introduce changes at some set of registrars. These statements could be more generalized to allow flexibility, but the principle of uniqueness is very important.

- Respect addition of security principles – share previous concern about adding scope or making this too complex for existing registrations. Do not want to risk over-engineering the auth-code. Other registry operators may have different desired security principles.
- Should there be a SALT that's managed by the registrant and by the incumbent registrar as well so that it's not reproducible?
- Did not say what the SALT could be so that there's flexibility. With respect about registry service providers, the hat worn when making suggestions is adapting commonly known security principles for one-time passwords.
- Can agree with all of these principles, but this would require a lot of work for registrars. Not sure how many other registrars will have the development resources available to get this done. Are we looking at this like best practices?
- From a security point of view, there are certain things that ought to be true if you're going to do things right. It is important to balance this with the cost of getting this done.
- There was some support for using principles so that the group could talk about discuss ideas and potential answers to charter questions. If these principles are not acceptable to the group, small group members could suggest other principles. At this stage, is there a minimum set of principles the group can agree to?
- If we consider utopia, we can talk about where we are now. Is it possible to look at what are two or three different principles that exist now at com, net, or biz and look at what exists today?
- The most important thing is that we have something workable. The most important concept is that it's secretive and safe. The main issue is how to keep the code secret for the registrant and safe for the registrar.
- From a registry point of view – in principle, it needs to be safe and secure and not visible when it's used. A registry should not be aware of the TAC unless there is a transfer in progress. The operational model should be that TACs are created on demand – they exist and then go away. If you create it on demand, there is probably better advice about creating it. For example, for small registrars, we should think about what we can offer to people to calculate a TAC so that it meets all of the principles that you want.
- TAC should not be retrievable from the registry – there is no reason to store a TAC – the operational impact on registrars is that they create it on demand. Should ensure it is not available use. The most fundamental change is to not give the auth-code to the registry unless it is intended to be in use. Also, registries do not give the auth-code back to the registrar. Other than that, no other changes need to be made.
- Do you consider the auth-code being held by the registry as a security risk? If that is the only change being requested, do not see a big benefit there either. In terms of dumping the auth-code as soon as it's provided to the registrant, that may not work b/c it needs to be provided to compliance. 1/3 of transfer inquiries involve the auth-code not working, and if it is disposed cannot do this.
- Think of an auth-code as a password. It's a bad practice to store passwords internally. You always store a hash of it. If a registrar is compromised, you don't want a bad actor to start transferring a bunch of domains. Even if it's not a requirement not to store the auth-code, should have a principle that tells people to move toward a model of the auth-code not existing unless they're needed. That is the best and most secure model for a registrar. If it's not stored, you're reduced the threat vector.
- Some customers are peculiar and want their auth-codes and want them in their system as soon as they transfer in their name. These customers want their auth-codes and we need to decide if we want to continue to allow that for their own safety. There is value in providing reassurance to the customer.
- Maybe the test here is have we reasonably captured the principles that we want to bring back to the WG.

HIGH-LEVEL PROPOSED PRINCIPLES:

- Should only exist at registry when a transfer is in progress
    - This suggests that regardless of when a registrar creates a TAC, it is not passed to the registry unless a transfer is in progress
- Must be unique per registrant and per domain name
    - The value must be safe and secure and not reproducible outside of the registrar, i.e., one-time password
- Must not be retrievable from the registry
    - A registrar can set it but a registry will never respond with it
    - A registrar would set a "NULL" to remove it from the registry
- Registrars should manage any TTL scheme

- Team's next meeting is scheduled for Wednesday, 21 July.

- For the final bullet points, these are topics that will be covered later in the charter, so it may be preferable to put those in parking lot for the moment and applied down the road.

- It may be helpful to think about what the small group would like to achieve between now and Wednesday – perhaps – could there be updates to the high-level principles? If everyone agrees, what are the implications for responding to the Charter questions?

- Are these principles going to be requirements or best practices? Answer – that's ultimately an open question.

- Action Item: Auth Code Small Group to review the high-level principles and consider whether these should be requirements or best practices. For example, what impact would these principles have on registrars if they were to become requirements?

- Action Item: Auth Code Small Group to consider the Status Quo discussion; for example, is there a next step to take with this that could inform the principles discussion?

Notes from Wednesday, 21 July 2021 at 14:00 UTC

- Recap from last meeting: different opinions on whether AuthInfo Codes as they are currently are acceptable, or whether additional security measures are necessary.

- One of the main goals in making any changes to the AuthInfo Code is whether it's possible to eliminate the FOAs (Gaining and Losing), if so does it make sense that there is some change to the AuthInfo Codes, even just standardization across Registrars.

- One of the things that we need out of today's call or if there is one more before next week is what message do we want to convey to the WG.

- Some registrars indicated that they wouldn't consider getting rid of the FOAs without some added security measures for AuthInfo Codes. What would those measures be?

- Important that the primary objective is to minimize the chance for unauthorized transfers, while keeping the protections for the Registered Name Holder (RNH).

- Also, can the Small Team develop a proposal that could help to answer the charter questions and/or other questions?

- On the alternate process proposal from Volker Greimann: In step 4: Either: Registry informs losing registrar of transfer request. Losing registrar provides auth code to registry, registry verifies auth code, registry conforms valid auth code to losing registrar OR: registry verifies auth code against auth code previously provided by losing registrar, Registry informs losing registrar of transfer request. Question: Does this suggest that the Auth Code is never sent to the registry? That the registry doesn't hold it at all? Answer: Two alternatives – 1) provided to the registry OR 2) already on file with the registry. Second option is more secure. It is worse if the registry doesn't already have the Auth Code on file.

- First option (storing at the registrar) would involve more changes, so not desirable.

- Another issue on 4a is that the losing registrar seems to have too much control over the process and could introduce delays. Anything thoughts on 4b?

- 4b is a better option where the registry verifies the Auth Code. When the registrant requests the Auth Code that is the point when the registrar could stop the transfer.

- The losing registrar should always have the Auth Code.

- Good for the registrar to keep the Auth Code for a certain amount of time.

- Was there another intent for the Auth Code? Seem to recall that there was something tied to registry lock – other aspects of the management of the domain that the Auth Codes match in the registrar and registry.

- There is also one on contacts so that you can move them.

- There are a couple of uses. Can get more info back in an inquiry if you have the Auth Code. If you send in the Auth Code for the domain you can get full contact information, but not sure whether this is done. Not familiar with a registry lock being used with an Auth Code.

- Need a system is compatible for users around the world. Auth Code is still the best solution. If we make the use of the Auth Code more ubiquitous the more we lose the security. Limit use to transfers and maybe for verifying requests for contact information.

- How about someone coming up with their ideal of what the Auth Code would look like? X number of characters long, etc.

- The registrar creates it and it is hashed at the registry – the registrar holds it or not, it's their business decision/security. It's not stored at the registrar and stored at the registry. We have a lot of comments about what users can or cannot do: Could we put in some language about 2-factor authorization by whatever registrar believes is secure. Make the Auth Code as long as possible. Registrant will cut and paste it anyway. Make it 32 characters with

upper and lower case characters with dictionary words not allowed. Random numbers, characters, special characters.

- On uniqueness – consider at least unique at the registrar.

- When you look at the wholesale market – don't want to have all of the security answers for each customer. It should be an option that the registrar holds the code so that they can support their customers. We can probably agree on the possibility of the registrar deleting it once they provide it to the registrant and registry, but should be an option for them to hold it.

- 2-factor could be provided via a number of different options.

- The ability to collect up AuthInfo Codes in order to do a bulk transfer – having it stored at the incumbent necessary may be necessary in order to support bulk transfers. The WG will look at partial bulk transfer. So the Small Team should look at storage of the AuthInfo Code. Looking at registries' codes it seemed the minimum of 16 characters was typical. 32 was a bit beyond.

- Seems like a good point (16 characters) to discuss in the full WG, or to make it flexible in a range – such as 16-32 characters. Registrants may just cut and paste, but some may have to type them in due to technical issues.

- Re: uniqueness: If we make this unique by domain name and registrant then for bulk transfers you would need multiple codes.

- Re: compliance: Should there be a check on whether the AuthInfo Code is compliant. Jody: This should be at the registry with a requirement for a minimum length and composition.

- There should some sort of registry check on the requirements, but would like to hear from a registry as to whether this is feasible. Could keep as an option to bring back to the group.

- Question: Since the Temp Spec was put in that broke the gaining FOA, curious what happened before Temp Spec for validating the Auth Code? What exists today? Jody: The registry is just verifying the Auth Code in the transfer request based on what they have stored. Just comparing what the gaining registrar has based on the registry's database.

- So just a comparison, with the option to hash it.

- The failure or success of EPP commands will often indicate if the Auth Code is valid. Believe there are some registries that will assign an Auth Code that conforms to their requirements for uniqueness and the registrar can just retrieve it. The ability to retrieve it will be important. If a registrar sets one and sends it to the registry, they may receive a failure notice that the criteria wasn't met for uniqueness. This would be set at the registry.

- The only reason we are discussing length/uniqueness is so that someone can't guess it for a fraudulent transfer. No matter how long or unique the code is if the account gets compromise they can still initiate a fraudulent transfer.

- Probably one of the biggest failure of transfers is that a registrar account was hacked or the email was hacked. Gaining access to the registrant's account at the registar.

- One thing we haven't discussed is enforcement at the registry. Some type of timeout with a failure to initiate transfer.

- There is a transfer lock and I wonder if those could be considered as to whether one could request to update the Auth Code.

- Think we have an outline to throw together. Should we meet again this week or doing this over email?

- Think we should have another meeting as we might not be able to finish via email.

- Seems clear that we aren't going to get to final outcomes on this issue until the WG deals with losing/gaining FOAs.

- Next meeting: Monday, 26 July at 14:00 UTC.

Notes from Monday, 26 July 2021 at 14:00 UTC

Notes

- Goal: Decide what to present to the WG and who from the Small Team will present.

Principles:

- Jim's first principle is important to start with and take to the WG and also the strawman.

- Are we starting at the high-level principles? Yes, good to start, particularly the first one: "Transfer Authorization Code (TAC) is an Identity Credential that upon presentation by a registrant identifies that registrant as the owner of its corresponding domain name."

- Re: "Should only exist at registry when a transfer is in progress. This suggests that regardless of when a registrar creates a TAC, it is not passed to the registry unless a transfer is in progress"

- Disagree: This endangers registrants in case of registrar failure. To the contrary, we should recommend that the TAC is immediately provided to the Registry upon creation, transfer, update. There might be different business practices. All kinds of customers that require possession of the Auth Code before they consider a transfer.

- Relying on the code being set early doesn't address registrar failure.

- If an Auth Code is set at the time of creation but isn't used, it could be hacked/be insecure. Does that outweigh setting the Auth Code from the outset? Could document what the reasons are to set the Auth Code or not.

- The overarching principle is: If you accept the first principle that the Auth Code is only for transfers, then the other principle is to treat that Auth Code/TAC as a one-time password. Adopt principles that are aligned with the concept of a one-time password. If you are going to bring security principles on the way things work, how do we ensure that there is a secure transfer process going forward? What can we do to make this better? That is where the other principles come from.

- The AuthInfo Code should not be stored anywhere. If you need a new one then you ask for it and get it. As soon as you store it then it is less secure.

- The high-level overarching principle is that the AuthInfo Code only for transfers. If it has other purposes we should document them. Does everyone agree?

- When we say that the AuthInfo Code should only exist in the registry at the transfer. This implies that the registry doesn't have it. It is too risky for the registrant – too much power to the losing registrar.

- Does appear that this principle would cause more stickiness, are there processes to eliminate that?

- On the topic of AuthInfo Code – if we are focusing on a TAC only for transfer, then we should still document other uses – such as proof of ownership of a domain that is unique. A password for a domain name. There currently are a lot of barriers in place to get Auth Codes. Being able to retrieve it with a low barrier is important. Need to be sensitive to that fact that there are other uses.

- Important to document those other uses.

- Good point that the AuthInfo Code can be used as proof of ownership – but that is consistent with the first principle, that the code does exhibit ownership. If you want proof of ownership for reasons other than transfer, you could give the password to the registrar account?

- We should make a list of what people are using it for, what is the risk/benefit, and do we want to change use of Auth Code to something else.

- The registrants are the customers of our resellers. Some are very protective of their customers and some want to be able to switch registrars quickly. This would require possession of the Auth Code by the resellers.

- Best practice would be that you eliminate as many of the storage of the Auth Code as possible.

- Re: "MUST NOT be retrievable from registry" – Why can't we make this that it can be set by the registry? Preserve the proof of ownership piece, but also allow the incumbent registrar to furnish the Auth Code to the registrant but allow the registry to determine it's valid.

- If the registrant needs it again, you could give them a new one – not the old one.

- Having the Auth Code not be stored at the registry, as soon as you put it there you make it eligible to move. If you only set it at the registry except when a transfer is in process then

you maintain control at the registrar. The registry should not have that control. But you could make it an option.

- We could present that the AuthInfo Code should be for transfers only but document that there are other uses, asking if those should still be allowed and/or when.

- One possibility is that we take chunks of this small team document, clean them up, and move them into the full WG document without the comments – "small team proposed principles for discussion". Any objection to this being brought to the WG in its current form? Could be helpful to have the comment summarized or to keep the comments.

- The agenda will be going out 24 hours before tomorrow's WG meeting. No time to summarize. Jim could present and other small team members could raise points/comments. WG members also will have access to the document, notes, and recordings.

- Could we Poll within the greater Transfer WG:
  Are registrars (or resellers) using Auth-Codes for things beyond initiating transfer? Ex: use of auth-info for proof of ownership or pre-transfer.
  Should the auth code be retrievable from the registry by the registrar of record?

Strawman List:

- The registrar creates AuthInfo code and it is hashed at the registry.
- It's not stored at the registrar and stored at the registry.
- Two-factor authentication, not necessarily using cell phone number, could be a security question.
- 32 character min. [Alternative: 16 characters]
- Require upper case and lower case letters, numbers, and special characters. Prohibit use of dictionary words. +Homoglyph consideration (0 vs O).
- Registry check to ensure AuthCode meets minimum requirements -- to discuss with the full group.
- Timeout after a certain number of requests to initiate a transfer.
- Can Transfer Lock status affect requesting/updating auth code in addition to just blocking a transfer request?

- What is the "Timeout after a certain number of requests to initiate a transfer." Prevent people from trying to guess the Auth Code after several requests. Let's say you get 10 transfer requests in one week then you don't allow more after that. Not sure how it would work. Hard to lock out, but maybe an acknowledgement to the losing registrar that someone has tried to transfer X times this week.

- Over a certain threshold the update of an Auth Code – You want to find a balance between those with a safe small threshold with two digits within a given time span. The issue with setting rate limits the bad actors will figure those out.

- Two comments on timeouts: 1) what problem are we trying to solve here? What benefit are you providing? 2) the suggestion that the registry should be evaluating a registrar – then you are involving the registry as a mandating reporter. Are there other mechanisms for managing whether or not registrars are well behaved.

- Re: 1) – to address brute force fraudulent transfer. It is an outlier.
- We would welcome scrutiny of bad registrars.
- We have had customers sending transfer requests for the domain name they would like to have. Only the registry will see the failed attempts.
- Maybe could have the registry just generate a message to the incumbent registrar for failed attempts.
- Jody agreed to introduce the strawman.

ACTION ITEM: Staff to move the cleaned up principles and the strawman list to the WG Auth-Info Code Working Document to send along with the agenda for the WG meeting, with links to the recording/notes for WG members to review.