Identity without authority

Decentralized Identity Systems

@laugh1ng.m0nk3y - @jade.rabb1t.23 - @ox.head.826
[cicada [AT] iamcicada.com]
Version 2017.21.03.BETA.001

The Cicada Identity system is the foundation of our decentralized application platform (DAPP). The ID is a decentralized, privacy guaranteeing Human Unique Identifier (HUID) for every person on the planet that returns control of personally identifying information (PII) back to individuals, allowing for revocable role based access control to that data.

It uses **biometric markers as an input to public/private keys** and a **comprehensive layered system of reputation that is dramatically different from today's concept of reputation systems to defend against sybil attacks and crush fraud** in a way that is literally impossible with today's archaic systems.

Implementation details of the ID system, such as creating the keys, reissuing keys, automated lost password recovery and roles of the HUID in the larger project are detailed in the original Cicada project paper so we won't rehash all of the details here.

Instead we'll focus on feasibility of the concept as well as provide clarification on specific challenges to the idea.

Without a doubt, the ID system is the component that gets the most scrutiny and also the most doubt. That doubt mainly swirls around whether it's even possible to build a strong ID system with no central authorities at all.

There tend to be two critiques of the decentralized Human Unique Identifier or HUID:

- The system is potentially gameable, in that without a central authority to verify it, someone could exploit it to create multiple IDs, also know as a Sybil attack which they can then use for Sock Puppet attacks
- It is not provably correct that it's a human, which is a variation on the first critique

The most common refrain was that it's "impossible" mostly because it's never been done and hence people have no frame of reference to compare it against.

We agree that a decentralized ID system is very hard but we don't agree that it's impossible. There is a big difference between impossible and very hard. It's impossible to flap your arms to fly. Whereas putting together a rocket that flies into space is very,

very hard but possible. Something like Zooko's Triangle was considered "impossible" until Satoshi proved that it wasn't with Bitcoin.

We've set out to solve these challenges despite their difficulty. Starting out is no guarantee of success. Many people set out to prove Fermat's Last Theorem for 358 years and failed, but if you don't start with the idea that it's feasible you absolutely will not succeed. That failure can also lead to additional success. Many of the attempts to solve the challenge resulted in advanced in mathematics, including the development of algebraic number theory and a proof of the modularity theorem. When Andrew Wiles started working on the proof for Fermat's theorem in secret, he had more reasons to give up than to keep going. Nevertheless, he kept pushing forward and seven years later did the "impossible."

Frankly, the human mind likes to label things as impossible because it's easier. If something is impossible that means there's nothing to do. No effort is needed because it simply can't be done. But if something is just very hard, that would mean someone has to work to bring it to life. Most of us are more comfortable with existing solutions or iterations on existing solutions rather than dreaming up new ones. There's good reason for that from a survival standpoint. We're wired to survive and reproduce. Thinking about hard problems burns a lot of energy. If you need to fight your way out of a burning car or dodge a snake you'll thank your brain for saving up as much energy as it could. But when we take that desire to conserve energy too far we're limiting ourselves.

The Cicada project sees the only actual flaw in human thinking as refusing to embrace what is possible. If we don't push ourselves, go further or reexamine our models of the world we can never make real progress.

The Urgent Need for a Decentralized, Distributed Human Unique Identifier

The Cicada project sees the need for a strong, privacy preserving, open ID that's decentralized and has no central choke points for one simple reason:

It provides the best protection against abuse, overreach, and authoritarianism.

Some form of digital ID for everyone on the planet is virtually inevitable in the near future. So either we create it ourselves, or someone will do it for us and we will not like the results. It's as simple as that.

The most likely creators of such an ID would be corporations looking to track customers or governments with values ranging from ruthless control to more open and fair and everything in between. Centralized organizations do not create vast information stores without a secret purpose. Whether it's a social media platform who's goal is to make money from advertising to you based on your personal likes and dislikes or a government that is looking to track, control, and surveil its citizens, those platforms will be corrupted from the start by those negative philosophies.

Those philosophies will get *baked in* to the final system and will remain inescapable.

- You cannot build openness into a system founded on closed ideas
- You cannot retrofit privacy or security onto a system built to subvert privacy and security

Privacy, openness and security must be foundational to the design of the system from the very start.

In fact, efforts are **already underway** in these areas. Banks and foreign governments are creating huge, weakly secure, non-revocable biometric identification systems in centralized databases that we have no control over. Companies like GenKey are working with the Indian government to create universal IDs for a billion of people, using a proprietary, copyrighted algorithm that people have no insight into whatsoever other than what they chose to share about it, which is not much.

If we allow governments or private companies to continue to create this technology, you won't know where or how your information is stored, whether the proprietary algorithms accessing it have good or evil aims, or even if it's really secure at all. You won't know most of this until it's inevitably hacked and all of your personal data spills out onto the nets, including your iris or fingerprint template or your DNA hash, which is now useless as an ID because you can't take it back or change it.

We feel strongly that these systems should be replaced with universally vetted, secured, and openly designed systems.

We simply can't allow this to be a private, closed system locked away from public scrutiny.

We also can't allow a central chokepoint to control it. The reasons for that are simple too:

Trust is *not* a fixed concept. Trust is a moving concept.

Trust can change easily from positive to negative or back again. If a man is loyal to his wife for fifteen years and then cheats on her, all his trust burns up over night.

When it comes to governments, they can go from liberal democracy to authoritarian nightmare in a flash. Leaving the controls of a such a powerful system to any one group is a disaster waiting to happen. If that group moves from trusted to untrusted, the game is over for everyone.

Thats why creating a decentralized identity system is not just a useful idea, it's a moral imperative.

The founding fathers of the United States created three branches of government, with their checks as balances, because power corrupts and absolute power corrupts absolutely. They were attempting to decentralize power as much as possible with the tools of their era. The system's design was intended to make sure that congress, the president and the courts could not move unilaterally, as is common in authoritarian regimes, where a few powerful people control everything with no regard for the people they rule. There are no good authoritarian regimes. Authoritarian regimes are only good for the small circle of people who control them and terrible for the other 99.999999999% of the people subjected to them.

The founding fathers used the best ideas and techniques of their time, a social contract of laws on ink and paper. Today we must use the best tools that we have available: digital ones. Without an updated, modern system of checks and balances we have no chance of preserving freedom and privacy in our increasingly digital futures and that should be completely unacceptable to each and every person on the planet.

Decentralized systems with checks and balances are the ideal way to ensure that no one entity accumulates too much power.

Human history has demonstrated again and again that **centralized trust is an oxymoron**.

The Catch-22 of IDs Today

The UN created the ID2020 initiative to give everyone on the planet a digital ID. It has noble goals, such as seeing an ID as a non-revokable right.

Many, many, many people in the world do not have ANY prior identification. According to the World Bank 1.5 billion people have no ID or access to get one. Even in a sophisticated economy like the United States millions of Americans have no ID. They have no birth certificate or anything else which makes in incredibly hard to get any form of ID. That's actually why the UN created their ID2020 project. It's the most laudable goal of the project, which is the idea that every person has the fundamental *right* to an ID, regardless of whether someone bothered to create a piece of paper that says the do at the moment of their birth.

Nobody should be able to say you don't exist and cannot participate at the most basic level in the system.

Yet with today's systems, you need an ID to get an ID, which is a catch-22.

Someone has to say you already exist as a person before you can be issued an ID. You need a birth certificate or a SSN or something else. But what if you don't have any of those things? You can easily be denied other rights like the ability to vote despite having lived in a place your entire life. You can prove you exist by, well, simply existing, but that does not mean you can get an ID in today's society, because someone else holds the key to allowing or disallowing you to have one.

A decentralized biometric ID, if properly implemented, holds the promise of giving every person a singular number that can't be taken from them, that uniquely represents them and that can't be denied.

The Cicada Project sees the HUID as a basic building block of a chain of IDs. It is a foundation on which other IDs are layered to add "weight." We agree with the ID2020 project that everyone has the right to an ID that is solely theirs and that can never be taken from them for any reason. You might rightly deny a person their right to participate in a society because they committed a crime like murder but they still exist as a person and that ID should remain theirs, despite privileges being stripped from that ID. The Cicada system allows for that natural complexity to exist just as it does in real life.

However, we diverge with the ID2020 initiative on a number of fronts.

Take a close look at the site and you'll find no mention of privacy or protection from tracking and abuse. Instead, the project is looking to partner with tech companies that will have a vested interest in making a closed system with secret agendas or at least a system they control which directly leads to abuse.

If a proprietary system is created with hidden, built in weaknesses and that ID becomes the default for billions of people who have no voice to object, the results will create slavery, not freedom.

An open, decentralized system can mitigate many of those issues but it creates it's own challenges as well.

With that in mind, let's examine the decentralized ID system and its implications. We'll take a fair look at legitimate criticisms of the decentralized approach and we'll look at how to deal with them.

But before examining decentralized IDs, it's helps to look at centralized IDs as they're more firmly established and their strengths and weaknesses are easier to pinpoint.

Central ID Systems Strengths and Weaknesses

First, we need to examine the assumption that centralized systems are automatically better or less prone to fraud, corruption and abuse.

If we look closely we can see that every system is gameable in some way, including centralized ID systems. This stems from the fact that no system is truly centralized anyway. Every system, even if it's created by a strong central entity like the US Department of Motor Vehicles, is still run by different people, with different moral systems and ethics, across a distributed geography. That's why multiple DMV employees have gotten caught for selling fake IDs over the years, subverting the very systems that centralized authorities created to establish trust and authority.

And that's just the ones that got caught. How many didn't get caught? How many forgers are out there making fake driver's licenses as we speak?

While it has certainly gotten harder to make a duplicate or completely fabricated ID since the 60s or 70s, when IDs used all analog technology, it's most definitely possible. In essence, every system is dealing with the problem of decentralization in some way, often without even knowing it.

That means that a central authority is no panacea to the problem of fraudulent IDs.

Centralized ID systems have the following two characteristics:

- **Strong controls at the input (creation of the ID)**
- **Weak controls on the output (use of the ID)**

The Social Security Number (SSN) in the United States is an example of classical ID system, issued to all babies at birth and designed to be unique. The biggest problem with the number is identity theft, which is a problem of *weak controls on the output side of the equation*.

The most famous early example of SSN fraud was in 1938 when a sample Social Security card was inserted into new wallets created by the E. H. Ferree company. The vice president of the company had the brilliant idea of using his secretary Hilda Schrader Whitcher's actual SSN on the sample card. Since then, at least 40,000 people have claimed Mrs. Whitcher's SSN as their own. It's maybe the world's first example of poor identity protection via human error but certainly not the last.

That's just the tip of the iceberg. The odds are currently 1 in 7 that someone has your SSN.

Even better, *government agencies often know this is happening but don't tell you*.

If your ID leaks in a corporate breach a convoluted set of rules may prevent the company from telling you until your personal information has been in the wild for months or years. Nearly five million SSNs are attached to at least three people.

Currently, our approach to security in the digital and analog world is to hand our ID over to multiple centralized authorities and trust them to protect it. If you buy a rifle in California, in the United States, they take your fingerprint and store it in a locked cabinet. If California legislators have their way soon you'll need to store that information in their centralized database online as well. Now, based on your politics, you may think that's a fair requirement and you may or may not be right. However we're not talking about whether it is a good idea *in the abstract*. What we're talking about is whether that government agency can maintain the security of your personal information. The answer is almost certainly no.

Just about every major government agency charged with protecting US citizens' information was compromised in some way in 2016, including the U.S. Department of Justice, the Internal Revenue Service, the San Francisco Municipal Transportation Agency and the CIA, as well as numerous very well funded mega-coporations and tech companies that should have access to the best tech security people in the world, including Oracle, Linked In, Verizon, DropBox, Yahoo, and Cisco. Also, don't forget the international SWIFT banking system which suffered an $81 million dollar heist. And that's just 2016. If we go back further it gets much worse.

Modern IT is a house of cards. It's built on layers and layers of highly complicated systems with hundreds of millions of lines of code. It's not hard to see why these systems fail us.

Think of all the interconnected challenges facing organizations today:

- Is all of their software up to date against known vulnerabilities, much less unknown ones?
- Do they have a system in place to rapidly patch all their systems?
- How often do they patch?
- Can they detect a breach after the fact?
- Is their firewall set up correctly?
- Do they have layered intrusion detection systems?
- Do their IT policies encourage flexibility and quick thinking or do they stifle the ability to act fast and deal with threats?
- How good is their IT personnel?
- Do they have the budget, know-how and skills to secure it?

In short, can they keep those system secure?

If history is any teacher the answer is no way. If you look at the last link you may be surprised at just how many of the companies and organizations you know and use are on that list of the largest data breaches of all time. From communications infrastructure companies like AT&T to tech companies like Netflix, Twitter and Minecraft, to

nearly every major bank like JP Morgan Chase and Morgan Stanley, to police agencies, health care providers, and court systems.

Anyone who has worked in tech for more than a few years knows that almost every system on the planet is currently vulnerability to sustained, advanced persistent threats and many not so advanced ones.

Centrally storing all of our IDs has proved to be a total disaster for society. It's no surprise really. Why rob a man when you can rob a bank?

You may have to rob tens of thousands of people to make millions of dollars but with one strike on a bank, you might just make off with millions in a single shot. The same is true for hackers looking for the big score in IDs to sell or use for fraud. It's better to hit a huge, poorly secured government database or websites and steal millions of IDs versus trying to hack lots of computers. Every day another major data breach sees millions of dollars lost, lives ruined and nothing ever changes. The average data breach now costs upwards of 4 million dollars.

That means that despite supposedly strong controls on the creation of centralized IDs, we still see massive fraud, whether through rogue DMV employees, human error, hackers or poor security and that's just a few examples of the broken system we all currently enjoy.

So when it comes to centralized IDs vs decentralized IDs maybe the question isn't whether it's possible or not?

The real question is can we do any worse?

The answer is no. In fact, we can do better for one reason:

All of this fraud and abuse *is due to weak output controls*.

So the solution is to flip the equation on its head, which is exactly what the Cicada system does. Let's take a look at how.

Cicada Decentralized ID System

The two keys of the Cicada ID System are:

- **Revokable biometrics**
- **Reputation banks**

Working together, these two powerful forces can provide for a much stronger system than the current centrally controlled IDs of today.

Decentralized ID systems have the following two characteristics (the opposite of centralized ID systems):

- **Weak(er) controls at the input (creation of the ID)**
- **Strong controls on the output (use of the ID)**

Biometrics represent the input control. The output control falls to the Reputation Bank.

After studying the critiques of the original Cicada paper, we noticed a pattern that pointed not to the design of the system itself but to the way we explained it.

- We realized we were thinking of a reputation system in a much more expansive way than its current definition and so there was a misunderstanding of it in people's minds as to what we meant by that reputation system
- It was not completely clear how biometrics help against Sybil attacks

It is the combination of biometrics and reputation banks that work against Sybil and many other kinds of attacks.

We'll examine biometrics first. Biometrics serve two purposes:

- Stop the average non-technical to fairly technical person from creating two IDs
- Allow you to prove after fraud occurs that you actually own the ID and stop it dead in its tracks

Deter Double ID Creation

Attacks are biometric systems are well known, but they are not in the realm of possibility for the average person. They act as a strong baseline deterrent from trying to create multiple IDs.

We chose iris scans because they have low false positives, they're fast and they're *hard* to game. Iris scan attacks are actually more challenging than the more well known attacks against fingerprints, like the infamous Gummy Bear attack or the Silly Putty fingerprint capture.

The only known current attack against iris scans are sophisticated. You must either scan an iris close up and get the template code of minutiae points or hack a database of

iris scans to get those codes. Then you use a genetic algorithm to iterate through variations on that iris code. From there you generate a high resolution image from that system.

Sound easy? It's not.

It's not enough to just crank out some images in Photoshop and hope that an iris scanner thinks they're real. The scanner won't think they're real just because you made up an Anime eye with your pen tablet.

The algorithms in the scanners are able to detect thousands of tiny intricaies in an eye, which is why these systems are used by Google and the FBI for facilities security.

As you might imagine, hacking a database, creating a genetic algorithm and printing out a high resolution image from that genetic algorithm are well beyond the capabilities of the average non-technical person and even beyond the capabilities of many technical people.

However, we're not naive enough to think that if the system increases in popularity more attacks won't be developed and become easier for people to use.

If the system is popular, people are incentivized to try cheat the system. Because of that more attacks will come to light and they will get easier to use.

For example, it wouldn't be hard for a person with the above capabilities to write a mobile app that someone could download and just press a button to get new eyes, provided they had a proper template image to start, which might even be their own eyes. Another example is that cameras might get so powerful that taking a snapshot of someone's eye close up or from a reasonable distance is enough to feed a 3D printer or other common device in the future to mold fake eyes that can be run through a scanner.

So why bother with biometrics at all? Two reasons.

The first reason is that today to get an ID it's not enough that you simply exist. It's the catch-22 mentioned earlier. You need an ID to get an ID. Of course, this is absurd.

With the Cicada system you do not need an ID to get an ID. You have one built in, your eye print.

You already have a unique identifier so why not use that as the basis of the system?

The second reason for biometrics as the basis for the HUID is that a biometric input acts as a proof *after* fraud as been committed against you.

Let see two examples of successful attacks:

- You're violently attacked and someone manages to capture you, put you under anesthesia, scan your iris and force you to reset your ID with a password they control
- Someone manages to compromise your private key and intercept your password by attacking your mobile device

In both cases an attacker would turn to these attacks because standard methods of compromising today's systems would be closed off to them, such as guessing passwords or using brute force attacks.

Even sophisticated attacks from afar are cut off to most attackers with the Cicada platform. For example, nobody can just scan your eye and generate the same private key without knowing your password. That's because the algorithm for generating it uses salt to randomize the output. No two private keys would be the same from the same eye.

At the same time a person cannot simply keep creating IDs from the same iris input, as outlined in the Cicada design paper, as a non-ID linked template of the iris minutiae points is stored in a separate blockchain and checked against during the time of creation. In other words when you create an ID, the blockchain returns a boolean yes/no on whether the iris template was already used somewhere at some time. No iris can be used twice, unless the proper password reset procedure is initiated, which generates a new public and private key and sunsets the old one, as detailed in the paper.

The system also enforces mandatory strong passwords via the blockchain. Those passwords are checked at the time of key creation against a blacklisted rainbow table of terrible passwords such as "password" and "password123" and any number of other ridiculous examples that non-technical and even technical people are prone to using because they're easy to remember.

That means an attacker would need to turn to malware or violence against you. Violence is the most difficult attack to prevent in any system. If you've ever seen the famous XKCD cartoon of the crypto nerd's vision of reality versus actual reality you understand this intuitively. Beating someone with a five dollar wrench until they give up their password kills even the best security. However, this problem *is not unique to decentralized IDs as centralized IDs are vulnerable to it as well, so the attack is a wash in the debate of centralized versus decentralized.*

However, we do offer one potential solution that no current system can offer. We are considering including a "coercion" password as well a regular password, which can be given under attack. After that the system proceeds as if you have given the correct

password for all transactions but those transactions can be rolled back at a later time. However, there are some challenges here that we have not yet overcome. The coercion password has the potential for fraud, such as to buy goods and then not pay for them, so the project is still considering ways to mitigate that challenge.

The second attack is also not unique to a decentralized ID system and it yields a better ROI for the attacker than violence because malware is ever present and they can hit more devices *simultaneously*.

However, we look to mitigate many of the current problems of malware *by using malware writers techniques against them* with a software checking blockchain. For example, many malware systems boot up and check against a remote system for commands before starting their normal operations or to see if they're running in a virtual environment with no network access (as a defense against anti-virus researchers). If they find violations of their rules they will not start. In a similar way the Cicada mobile application will not allow the app to boot on a local device if it has the wrong signature or the wrong datestamp or version of software or if it has no network connectivity, etc.

Neither type of attack is possible to prevent 100% of the time in any system, so a good system takes this into account at design time.

Let's assume that in both cases the compromise does happen.

After the attacker happily begins to assume your identity, using your cryptocoins, posting crazy things on social media and applying for loans, you are now the unhappy victim of identity fraud. What can you do?

In today's world identity fraud controls are seriously lacking. For instance, special alerts go on your credit cards, with those alerts run by a central entity like Visa or Mastercard. When someone opens a new card in your name it's supposed to trigger an alert and a call to the victim. But it doesn't always work. More often than not the attacker gets away with numerous attacks before moving on to another victim. They open new credit cards, buy a bunch of things and never pay the bill. They do this dozens of times. Even if some of the attacks are thwarted, a few succeed and they get away with their crime.

So how does your built in biometric marker fix this problem?

Imagine that the Cicada system has grown large and ubiquitous so it is widely used by countries, companies, schools, non-governmental organizations and more.

Two types of recovery are possible when the system is large enough:

- **bonded proof of stake private companies**
- **state or federal courts**

Now all you need to do to recover your life after identity fraud is to prove that you actually have the iris in question by showing up in court or at a private facility. Your iris is scanned and it's matched against the homomorphically encrypted biometric template blockchain with a boolean response. It either exists or it doesn't. If it exist you are the only possible owner. That means that any ID that is created from it would be yours. Now those two entities, along with you have the collective power to *flag that ID as dirty and blacklisted* and allow you to create a new one that resets your life. This works like a the two keys needed to turn a missile launch The process would have to create a new key and then your new key plus the verifier would allow overriding the compromised key and flagging the compromised key as blacklisted.

The creation of these entities is inevitable as more and more people come to rely on the system. In the early phases they will be bootstrapped with a bonded proof of stake dispute organization organized by the Cicada creators, governed by a distributed board of stake holders to help with small scale disputes and early problems. These can be community based and volunteer organized. As the system evolves the structure can become more codified.

This is infinitely more elegant and simple than the current broken system.

Reputation System

However, rather than looking to deal with fraud after the fact, the ideal system stops fraud before it occurs. Biometrics are only a deterrent. They will not stop people completely from making multiple identities.

But we don't have to prevent every fake ID from being created.

Instead the real advantage of the Cicada platform comes from *preventing that identity from being used effectively* on the network, something that no system in use today can claim with a high degree of certainty.

The Cicada system works to prevent fraud on the system with an innovative and expansive Reputation System (RS).

When most people think of reputation systems they think of the Wikipedia definition which is "programs that allow users to rate each other in online communities in order to build trust through reputation." They picture Ebay seller ratings or Yelp restaurant ratings or Uber's driver and passenger rating system.

This is NOT how the Cicada project conceives of a reputation system.

While some form of people rating each other plays a tiny role as part of the larger reputation system, it is ineffective as a truly comprehensive trust mechanism. A system

that allows people to rate everything with stars provides only a baseline sanity check and it is subject to group think and vote trading, as well as making it hard for new, legitimate businesses to get a foothold because they have no rating, which penalizes them unnecessarily. A fantastic new restaurant might have no ratings and so struggle to find customers while a moderately good but long standing competitor dominates.

Reputation systems have MUCH more potential if they are considered in a more expansive way.

The Cicada project sees an RS as something closer to the definition provided in the book "What's Mine is Yours", by Rachel Botsman and Roo Rogers, about the rise of the sharing economy. The authors note that 'it is only a matter of time before there is some form of network that aggregates *your reputation capital* across multiple forms of Collaborative Consumption'.

However, we differ from the book's definition. The Cicada project argues that "Collaborative Consumption" is too restrictive.

Why wouldn't a reputation bank bridge all kinds of consumption, the non-collaborative centrally manufactured consumption of today and the potential collaborative consumption of tomorrow, as well as non-consumption activities like forming groups and communicating?

The Cicada project sees a multifaceted reputation system that exists at all layers of the stack from the network communications protocol, to commerce transactions, to voice and text messaging, to voting.

A Reputation Bank Blockchain (RBB) holds a HUID and sub-IDs Aggregate Reputation Score (ARS) which acts as a kind of virtual currency that allows you to interact with the system at various levels. The RBB acts as a trusted arbiter, along with various smart contracts on the network, to efficiently and effectively provide Role Based Access Control (RBAC) to various actions and groups on the network.

The ARS acts as a kind of "Turing Test" in reverse. It says "this is a person and not a sock puppet/robot/fake ID but an actual person who is allowed to do XYZ action."

In other words the system doesn't need to prove you haven't gamed the system at the point of ID creation only. Instead, your reputation is the guiding force on the network, which comes from how you act on the system.

Some things such as posting on a forum require little to no reputation whatsoever, whereas voting in a national election might require the absolute highest level of trust/reputation/RBAC approval.

The RS has multiple types of ways to gain or lose reputation on the network, which are:

- Algorithmic
- Weight based
- Ratings

Algorithmic Reputation

Algorithmic are the simplest forms of reputation on the system. They are also automatic. A good example is the Network Reputation System (NRS). Imagine a reputation system built into the TCP/IP protocol. It looks for anomalies and provides a universal sanity check for all nodes on the network. There are two blockchains involved in keeping the network safe.

- network-rules
- network-control

Think of the network-rules blockchain as a kind of version control for a combination of firewall rules, intrusion detection rules and sanity checks.

These rules govern transactions on the networks. This builds on concepts from Bitcoin as well as from enterprise network defense architectures and *builds them right into the networking protocol itself*.

For example, Bitcoin includes a baseline sanity check for transactions that come into the network. According to the book "Mastering Bitcoin" from O'Reilly press:

"Before forwarding transactions to its neighbors, every bitcoin node that receives a transaction will first verify the transaction. This ensures that only valid transactions are propagated across the network, while invalid transactions are discarded at the first node that encounters them.

Each node verifies every transaction against a long checklist of criteria."

Some examples from the book are:

- "The transaction's syntax and data structure must be correct."
- "Neither lists of inputs or outputs are empty."
- "The transaction size in bytes is less than MAX_BLOCK_SIZE."

None of the nodes on the network trust that the rules are being followed just because they receive data from another node. Instead *each node does the verification itself* to ensure that incorrect transactions do not propagate.

With the Cicada project, the network-rules blockchain includes sanity check rules such as the above, as well as signatures for various attacks on the network and packet governing rules. They might indicate that a particular field should only contain X type of data or define the correct order of signals such as RST/ACK/FIN and flag nodes that are sending those deliberately out of order.

A flag in the chain also indicates the most up to date ruleset and the minimum ruleset number that needs to be running on the nodes or else it is not allowed to join the network at all until it downloads and runs the updated rules. This means that every node will have built in intrusion detection and attack signatures running by default which dramatically improves network security.

However, some people will try to cheat the system, either by disabling rules or building their own nodes that ignore the rulesets. That is where the first level of the reputation system comes in.

The network-control blockchain contains hosts flagged for behaving poorly as defined by the network-control blockchain rulesets. Nodes that are misbehaving either through deliberate attacks or through malware infection will be automatically rated via the NRS. Every node that is verifiably running the latest rules gets a vote on the network. Votes are initially pushed to a distributed hash table for temporary storage or a distributed DB/file system and then eventually pushed to/aggregated to the network-control blockchain. A challenge/response verifies that they're allowed to vote, proving they’re running the correct binaries signatures of rulesets.

A node rates nodes that it interacts with as their packets come in.

A rogue node that communicates with a good node that cannot prove that it’s running the correct ruleset gets an automatic hit to its reputation score. If that rogue node continues to behave badly after that initial hit, such as sending malformed packets, as defined by the ruleset, it continues to take hits to its reputations score. Eventually the good node may enact a temporary local block via its firewall rules, stopping all traffic from the rogue node. It will then vote to the wider network that NodeX is behaving badly. If enough votes are reached from the network the node may be blacklisted for escalating periods of time.

At first it might be five minutes and then seven and then ten and then fifteen. Eventually a node might be blacklisted automatically for days or even permanently until that node can prove the correctness of its network stack again through an automatic verification process against another blockchain.

Higher up the stack there are additional algorithmic reputation bumps, such as how long an ID or NodeID has been in use, the types of transactions it conducts on the network, and the amount of successful mining operations it has completed. The algorithm might look for interactions on the network, such as whether the node has joined any groups and participated in that group. These types of interactions automatically bump or drop

the reputation of the ID. These interactions require time and persistence. They cannot be faked.

Everything on the network can be thought of as transactions. All transactions have a built-in level or trust requirement. Whereas joining a local social group might have little to no trust requirements, exchanging or accepting money as a business or voting in a national election may have very high trust requirements, with those requirements built into the protocol itself.

This makes it hard for a person to simply spin up a fake ID or a rogue node and do damage. Instead they must "live" with the ID and node day in and day out, establishing a digital footprint of trust that attaches to that ID or they can't do anything useful.

Also, to be very, very clear, the system is not tracking all of the actions of the IDs on the network. Instead it is tracking the resultant score of that transaction. A node may interact with another node and violate multiple protocol rules, but the only thing recorded is the fact that the reputation is negative not the violations. In the same way, we will see that when an ID is used on the network for particular types of transactions, such as forming groups and interacting with people, the resultant reputation score is all that's recorded, not the transactions specifically. We'll examine that more closely in the next section.

Weighted Reputation

The next layer of reputation that more directly affects the biometric HUID versus the nodes that the HUID runs on the system are "weight" based reputation scores.

Positive weight is added to an ID based on interactions with that ID over time. Additional weight means that the ID is being used on the system in a way that is not hostile to the system. Weight is added through simple human interactions. In essence it is a kind of Turing test for people. In other words the system is asking are you a real person in the world?

It knows this by actions taken in the world. For example, let's say that you create a book club with twenty people on the Cicada decentralized app platform. You know fifteen of the people but not the other five. One of the people you do know interacts with you for the first time by sending you a chat message. Before the connection is allowed, you have to accept the interaction. By accepting the interaction you have indicated that you know or are open to communication from that person, which adds weight to their reputation score simply because you have indicated they are someone you wish to communicate with for now. You might also receive a random prompt that asks whether you know the person personally or "in real life" that you may choose to answer or not. This would add additional weight. However you are not "rating" the person as you would through Uber. They don't get five stars from you. Instead by interacting with them you "voted" for their "humanness."

Let’s say on the other hand that you receive a message from one of the five people you don’t know in the group. You may not be open to receiving messages so you don’t click yes. That leaves the ID as neutral to you and to the system.

Lastly, you may choose to interact with an unknown person only to find they're kind of a jerk, so you cut off the communication and block them, which results in a weight drop and a hit to their reputation for that level of transaction, communicating to strangers but does not prevent them from doing other things on the network they are allowed to do.

Generic rules define successful weight bumps and decreases for groups via smart-contracts or additional blockchain rulesets that are required to run on a client. For example, automatically spamming everyone in the group in order to get nineteen rate bumps would result in the opposite, a rate drop.

Eventually, groups themselves may define parameters for successful weight bumps or losses provided they pass a kind of bloom filter or a bayesian filter for criteria to ensure people are not simply making up crazy rules to game the system, as is often seen in massively multiplayer games where people find an exploit and quickly level up their character.

Naturally some folks will find a way to game the system, but additional updates to the rules framework will eliminate the loophole and deliver a reputation hit for cheating the system in the first place that should deter others from trying to do the same thing in the future. A governance board, elected through the Cicada system itself, will stand as the arbiter of those rules, and they will be subject to recall and replacement and not permanent fixtures. By contrast they are not to be simply popularity contest politicians so they will have some autonomy to ensure correct rules are propagated. The system users as a whole will have a right to vote on or against rules, by following the community vote process as outlined in the original paper.

There is also a second kind of rate bump in the system and those come from “trusted” centralized entities. Though the Cicada system is decentralized at its core, centralized entities will naturally develop on top of the system. This is similar to planets and other regularized defined structures developing from the chaos of the underlying universe. Naturally some of these centralized entities will hold keys to various “spheres" on the network.

Each group on the Cicada platform is called a sphere. A sphere has its own identity and can have its own rules for joining, as well as RBAC controls and administrators. A sphere could be a sewing circle, a corporate entity, and so on, all the way up to a nation state. Each of those spheres naturally go from little to no control to heavy controls required.

Let’s imagine that the system has reached a level of sophistication where people are using it as a way to get you onboarded to a new job at a company. Your ID gets you shares of stock, access to the VPN, access to certain documents on the network. It

also gives you certain rights and privileges in the company. Perhaps it also gets you access to your paycheck, which may come via cryptocoins to your private wallet or a shared bank wallet.

No company is going to let a sock puppet ID join their company. If you managed to figure out a new attack on the iris scan and Photoshop your way into some low level transactions on the system that is one thing. But building a fake ID to get a job and scam a company is another thing. As such the company will naturally vet you through various common techniques, such as meeting you in person, interviewing you, checking your resume and past employment. In other words they're adding weight to the idea that you're a real person and your ID is actually you.

When they decide to hire you, they deliver a significant bump in your reputation score by voting with their weight and adding it to yours. You are now gainfully employed by them. That bump might be significant enough that it allows you to do a number of lesser or equal transactions on the system in that you will be automatically at a certain trust level.

Ratings

Lastly, there are simple rating systems on the network, such as we see on Uber's passenger and driver rating system. However they carry a lot less water for most transactions than overall reputation. In order to avoid a Black Mirror like result where everyone is rating everyone else on everything from their smile and friendliness to their likability, ratings are confined to areas where they make sense, such as in business transactions where two parties are unlikely to know each other at all.

By confining a classical rating system to these narrow confines we mitigate the risks that people can game the system.

We categorize a "classic" rating system as any of the following:

- Like/Dislike
- Up vote/Down vote
- Star rating
- Note based feedback

These are the most common ratings systems in the world today and while they provide some level of confidence they're imperfect at best. They exist on websites like Yelp, Ebay, Reddit, Facebook, Medium and pretty much any other site that wants to incorporate user opinion into their platform. The reason they're imperfect is because they're subject to all manner of well known attacks such as:

- Paying people/companies to endorse you, aka "turking" = This attack is nicknamed after the Amazon "mechanical turk" system where you hire a group of

faceless people to crank out a meaningless task. This is essentially a brute force attack.

- Revenge rating = This is where someone deliberately rates someone poorly because of the feedback they got first.
- Defensive rating = This is where people rate people higher than they normally would to protect themselves from getting a negative rating in return (done all the time on Uber and Lyft such that the range of “good rating" is not 4-5 but 4.8-5 {an absurd range} and anything lower could mean termination).
- Sockpuppeting/botting = Here people or businesses employ an army of scripts to increase ratings through upvoting as we have seen recently on Reddit and Twitter.

All of these attacks stem from the same root:

They require active human judgement calls which at their core are badly flawed.

Because of that, these systems are prone to human bias and reasoning defects. For example, popular posts on Twitter or Medium tend to create a self-fulfilling prophecy where they get more votes just because they’re already popular, while more worthy posts might languish in obscurity because they’re not more popular. The fact that something is popular might have no bearing on its quality. It might simply be a matter of the author having a significant following so pure statistics mean that at least a certain percentage will like anything they write and hence it will get upvoted with a disproportion number of votes that in turn encourage other people to vote for it, despite its lack of merit. These systems can’t account for *actual* value.

These system are also prone to battles of ideology. Take any controversial political writer on Amazon, either on the far left or right and see how they gather up mindless numbers of five star and one star votes, from their supporters and opposition. None of those supporters or followers bother to read a word they've written, they’re simply casting a vote for their perceived team. They are “cause” voting, so again we learn nothing about the actual quality or particular merit of that work.

All of this makes classical reputation systems not very reliable but they are the best ideas we have come up with until now. To be fair, these are all outliers. For the majority of the people and companies on the inner parts of the bell curve, the ratings tend to be a rough approximation of quality and status.

In the Cicada system we look to eliminate as much of human bias as possible be removing the concept that someone is rating something as good or bad, which is essentially what an upvote/downvote or like/dislike is at its core. It is nothing more than an expression of preference. A better system is one where we simply give a plus or minus to something based on whether it was successful.

For example, let’s imagine we're rating a ride sharing driver as we do with something like Uber or Lyft today but on the decentralized Cicada platform. With current systems we give them “five stars” and maybe a comment. A better choice is to simply rate them on a few check boxes of categories. Did they drive well? Did they pick me up on time? Were they friendly or not? The choices might toggle through a pool of questions to randomly give people a plus or minus. Over time these should work as a better approximation of where a company, person or service is effective, reliable, fair and honest in their dealings with the rest of the world. The results add or decrease the overall reputation of the ID in question automatically behind the scenes.

Many of these checks can be automated. The system would know many things that are positive or negative, based on when the ride was called, such as whether the driver arrived on time or late. A latent plus or minus would get logged with the system that the human vote would work against or in concert with once they enter their feedback. If the automated check and the person agree than it might form a bonus. However, if the two disagree, than that may zero out or neutralize the votes in either direction. In this way the automated checks are balanced by the humans and vice versa.

Conclusions

We’ve seen how a decentralized ID system can provide robust security, by mitigating most forms of fraud, and allowing someone to stop fraud after it occurs by having built in biometric proof that the ID is theirs alone. We’ve also seen that a more forward thinking and carefully designed Reputation System (RS) can help to prove that an ID is real through its usage over time. The combination of these two factors allows us to create a strong, people controlled and open system that cannot be subverted by centralized entities that have become malicious.

A strong, privacy focused, decentralized ID can help overcome many of challenges we have today in the digital and real world but only if it’s designed correctly from the very start. The concepts of freedom, privacy and security must be baked in from the beginning or the system that we eventually get will be a system that nobody wants to live in.

We don’t have to wait for someone to build that system for us. We can built it ourselves. Nobody is coming to save us. As always it is up to the people to save ourselves.