Effective Team Dynamics in Botball and the Security/Privacy Research Field
Jeremy Rand
The Namecoin Project / Norman Advanced Robotics Alumni / Team SNARC Alumni
jeremy@namecoin.org / jeremyrand@danwin1210.de

Effective Team Dynamics in Botball and the Security/Privacy Research Field

1 Introduction

Although Botball might seem on the surface like just an engineering challenge, with the best mechanical and software designs winning, Botballers are well aware that another factor is team dynamics. Teams with effective practices are likely to produce better engineering ideas, and are also likely to produce better implementations of those ideas. I've been working in the security/privacy research field at The Namecoin Project for 13 years, and my experience with team dynamics at Namecoin very much mirrors my 13 years of Botball experience.

This paper aims to serve as a case study comparing the team dynamics in Botball and the "real world" of security/privacy research. It can be considered as "Part 4" of my series of GCER papers in this theme ("Making HTTPS and Anonymity Networks Slightly More Secure" [1], "From Hacking Botball Controllers to Having My Code in the Number One Privacy Web Browser" [2], and "I Won Awards for Hacking Botball Controllers; Now I'm Protecting the Anonymity of At-Risk Internet Users" [3]), but each paper in this series can be read on its own.

2 Project Case Study 1: Namecoin and TLS

2.1 Intro to TLS

You've probably heard that, HTTPS is the "secure" version of HTTP. What's actually more secure about HTTPS? The difference is that HTTPS wraps HTTP in a security layer called TLS. TLS performs two major tasks: *encryption* and *authentication*. Encryption ensures that only the web server you're talking to can read what you send; authentication ensures that the web server you're talking to really is the website you typed into the address bar. Encryption is the easy part; it's relatively well-understood math without major known weaknesses (except for quantum computers, of course). Authentication is trickier: how do you mathematically prove that a web server is who it says it is?

TLS solves the authentication problem by using *certificate authorities* (CA's). CA's are corporations who sign a *certificate* asserting that a given website address (e.g. https://www.kipr.org/) is associated with a given *public key* (essentially a large number). The website owner can prove ownership of that public key by using an associated large number, the *secret key*. Your web browser verifies two things when initiating an HTTPS connection:

1. The web server has demonstrated control over the public key it presents, using its knowledge of the secret key.

2. The public key it presented is signed by a CA as belonging to the website you're trying to connect to.

You might see a concern here: how do we know the CA's aren't lying? Well, we don't! And indeed, sometimes the CA's have been known to sign false certificates, either due to malice or incompetence. In such a situation, TLS loses its security guarantees.

2.2 Namecoin: Decentralized CA's

There are some parallels between TLS and money. In the same way that TLS loses its security guarantees if the CA's are compromised, money loses its security guarantees if the banks are compromised. And indeed, there's a math proof from Leslie Lamport dating back to the 1970's, which asserts the impossibility of removing that trust requirement.

Bitcoin [4], however, exploits a loophole in Lamport's impossibility proof, enabling individual users of money to "be their own bank". Namecoin [5], a spin-off of Bitcoin, utilizes the same loophole to allow individual website owners to "be their own TLS CA". By registering your website address in the Namecoin blockchain along with your TLS public key, anyone accessing your website can be assured that the public key your website presents really is yours. Reassigning your website's entry on the Namecoin blockchain to a different public key without your consent would be analogous to stealing your bitcoins – a nontrivial task.

3 Project Case Study 2: SocksTrace and Proxy Leaks

3.1 Intro to Tor

Tor [6] is an anonymity network that bounces communications through a series of volunteer-run relays to ensure that users can talk to each other without knowing each each other's identity or location. Tor utilizes a cryptographic technique called *onion routing*: your computer encrypts your traffic multiple times, once for each relay that your traffic will pass through. As your traffic passes through each relay, the relay decrypts one layer of the encryption (like peeling back a layer of an onion), revealing the instructions for which relay will receive the traffic next. Only when the final relay decrypts the last layer of encryption is it revealed which actual website you're visiting. Onion routing means that no relay knows the full path that your traffic took, and therefore no relay can determine who was visiting which website.

Tor has been successfully used by whistleblowers to prevent governments from detecting that they leaked documents regarding official wrongdoing; it's also routinely used by average people who don't want advertising companies building up a dossier on their interests, and by minors who don't want their abusive parents spying on their online activities, among many other use cases.

3.2 SocksTrace: Extra Protection for Tor

When launching Tor, you'll be greeted with this warning:

> Tor can't help you if you use it wrong! Learn how to be safe at https://support.torproject.org/faq/staying-anonymous/

This link contains a variety of warnings, the first of which is:

> Tor only protects applications that are properly configured to send their Internet traffic through Tor.

What does “properly configured” mean? Applications access Tor via a proxy. By telling an application (e.g. a web browser or email client) to use the proxy that Tor exposes, in theory you’ve “properly configured” it. Unfortunately, practice deviates from theory. Many applications exhibit bugs in their proxy support, e.g. they may send a small fraction of their traffic outside of Tor. This class of bugs is called *proxy leaks*.

SocksTrace is a tool (led by my colleague Robert Mindo) that detects proxy leaks. When leaks are detected, SocksTrace can block the leaking traffic, log details to facilitate filing a bug report, or even automatically redirect the traffic over the proxy. We’re also investigating making SocksTrace enforce other security policies, e.g. mandating TLS encryption. We’re working with projects like Tor and Whonix [7] to integrate SocksTrace into their software.

4 Project Case Study 3: Occlumask and Semantic Content Leaks

The *second* warning that Tor’s aforementioned startup link contains is:

> if you provide: name, email, address, phone number, or any other personal information, you are no longer anonymous to that website. The best defense is to be vigilant and extremely cautious when filling out web forms.

This threat is more subtle than one might imagine. For example, talking about the weather would violate this advice. “Be vigilant” is not really great advice, because it’s inconsistent with human psychology. The set of things one might say that could reveal their identity is called *semantic content leaks*.

Occlumask is a tool (led by my colleague Alice Margatroid) that scans text you type (before you send it) that warns you if semantic content leaks are potentially present, so you can edit the text to prevent inadvertently revealing your identity..

5 Parallels to Inter-Team Dynamics in Botball

Approximately all humans would recognize cooperation *within* your team as an important skill. But Botballers also understand that exchange of ideas between teams at events like GCER is crucial. Conferences are where researchers discover new leads, by talking to adjacent researchers. This is the case in the security/privacy research field too. Put another way, to some extent, all Botball teams can be thought of as subteams of the same larger team (the larger team’s goal being to maximize the

educational value of Botball), and all security/privacy research groups can also be thought of as subteams of the same larger team (the larger team's goal being the advancement of privacy tech).

5.1 Shared Knowledge on Efficiency

My KIPR Open/Aerial team (Team SNARC) presented at GCER 2014 [8] on Monte Carlo subsampling, a technique for dramatically improving the efficiency of the Asus Xtion sensor. Monte Carlo made it feasible to process the vast amount of data coming from the Xtion at approximately real-time, without hitting the CPU resource limits of the KIPR Link controller. Quite a few Botballers gained practical knowledge from that presentation that helped them with their own bots.

Meanwhile, one of the Tor developers, Q Misell, spent most of 2025 going down a rabbithole pertaining to train ticket security infrastructure. While traversing this rabbithole, she noticed that train tickets utilize an unusual, highly compact compression system for certificates, known as *PER*. (This makes sense, since the certificate needs to fit in a QR code that's low-entropy enough that the ticket can still be scanned even if it's physically damaged.) Namecoin also has storage constraints (the nature of a blockchain requires the data to accumulate and stick around forever). When I met up with Q at the FOSDEM conference in 2025, she suggested to me that PER might also be a good fit for Namecoin, to compress the data we're storing on the blockchain.

My colleague Sofia is now working on exactly that: integrating PER compression into Namecoin. Initial results appear quite promising, although evaluation is still ongoing. Swapping ideas with Q at FOSDEM enabled this efficiency improvement on our end, much as attending my Monte Carlo talk at GCER enabled other teams to do sensor efficiency improvements.

5.2 Shared Knowledge on Hacking

A particular interesting example of knowledge sharing at conferences involves `LD_PRELOAD`, a method of hacking software to behave differently without needing to modify the software itself. `LD_PRELOAD` works by intercepting the communication between the software and its libraries. I first learned of `LD_PRELOAD` via Dan Kaminsky's talk at Chaos Communication Camp 2011 [9]. Dan's talk covered using `LD_PRELOAD` for enhancing authentication security, including for TLS (other parts of his talk covered things like Bitcoin and censorship-resistance tech). I didn't have much use for `LD_PRELOAD` at the time, but a few years later, I successfully utilized `LD_PRELOAD` to hack the AR.Drone's camera, which I presented at GCER [10] [11].

What made this example unusual was that the topic then came full circle this year: we identified at least two situations where `LD_PRELOAD` would be helpful for SocksTrace. Thus, a talk related to Bitcoin and TLS, which I only remembered because I used its techniques in Botball, ended up being reused by Namecoin.

5.3 Shared Knowledge via Adversarial Competition

Botballers are well aware that the adversarial, competitive nature of Botball yields educational advantages. Being able to see how your code performs while other teams are trying to exploit whatever

weaknesses your strategy may have is a natural vector for finding ways to improve your code and strategy.

We recently encountered an example of a similar improvement opportunity with Occlumask. Less than two months after Alice presented Occlumask at the 39C3 hacker conference [12], a new paper came out from machine learning researchers at MATS, ETH Zurich, and Anthropic [13]. Their paper, titled "Large-scale online deanonymization with LLMs", could be considered the flip side of Occlumask: while Occlumask uses LLM's to protect anonymity, this paper uses LLM's to attack anonymity.

In the same way that Botball teams can derive improvements from going against their opponents' bots, we were able to utilize this paper to improve Occlumask. This also drives home an important point about security research: attack papers can be repurposed for defensive work. It's a *good* thing to study attacks, even if you're on the pro-privacy side: this is how the defense improves.

6 Parallels to Intra-Team Dynamics in Botball

Cooperation within your team is, of course, not to be neglected, and there are examples of parallels here too.

6.1 Task Assignments

Each Botballer has their own skill set and interests. A team's ability to perform well at diverse skills lies in part with the team's ability to synergize the diverse skill sets and interests of its members. In practice, this means that an effective team leader will keep a mental database of what each of their teammates enjoys working on and is good at. This enables the team leader to accurately gauge which members should be assigned to a new task.

We encounter this in Namecoin as well. As a recent example, SocksTrace is Linux-only at the moment. The Tor Browser Team had indicated that they'd like to see it ported to other operating systems like Windows and macOS. A Windows port of SocksTrace would involve using the Windows equivalent of `LD_PRELOAD`, which happens to also be a very common approach for hacking games on Windows.

As luck would have it, Sofia joined our team about a month before we needed to plan out our next SocksTrace grant, and Sofia had mentioned to me that she has experience with Windows game hacking. This led to the following chatlog excerpt:

> Jeremy: I figured out a potentially good project to assign to you. Spoiler: it is likely that your game hacking experience will be relevant.
>
> Sofia: oo that sounds fun
>
> Jeremy: [technical outline of SocksTrace Windows porting]
>
> Sofia: oooo this definitely does sound fun
>
> Sofia: you know me well lol

A major reason I was aware of Sofia's game hacking background was that it's pretty common for Namecoin developers to socialize by talking about computer science interests besides Namecoin; I also have some experience with game hacking (mostly GameCube games), so this was a conversational topic several times. Much as Botballers tend to be friends with each other outside of purely robotics-focused contexts, Namecoiners also tend to hang out in non-work contexts, with results that turn out to be beneficial inside and outside of work.

6.2 Practicing Presentations

It's pretty common for Botballers who are new to giving GCER presentations to practice their talks in front of teammates who have done GCER talks before. This has several benefits: it builds confidence, it improves the quality of the content, and it helps ensure that the time limit is respected. Students who may derive extra benefit from such practice include students with social anxiety or speech disabilities, students who aren't native English speakers, or students who just tend to talk fast (especially when excited or stressed).

In the "real world" of giving conference presentations for work purposes, the stakes are, of course, somewhat higher. The conferences that Namecoiners present at have a larger audience than GCER, and the quality of our presentations directly affects funding and collaboration opportunities. We therefore put extra effort into making sure that our newer developers have sufficient support in preparing presentations.

In one recent example, three of our developers arrived in a conference's city about a week early, and we met up in my hotel room for three sessions (several hours each), as we practiced presentations for each other. Everyone got progressively more comfortable with their slides over time, and was able to deliver talks with a more relaxed demeanor and with higher-quality ad-libbing of content that wasn't on their slides' text. For one of the developers, I borrowed a trick from well-known psychologist B. F. Skinner [14]: that developer is a huge fan of Warheads candy, but live in a country where Warheads are hard to find. So I brought a 4-pound bag of Warheads with me to the conference, and every time they made progress on their talk, they got a Warhead. This helped them stay focused and motivated, and they were able to deliver the talk at the conference flawlessly.

The organizers of that conference took note of how good Namecoin is at supporting our devs at giving talks. At a subsequent conference by the same organizers, one organizer commented to another that some other projects gave subpar presentations, and that "the problem those projects had was that they didn't have a Jeremy in-house to help coach their new developers on how to give awesome talks." The main reason I'm good at such coaching is due to Botball.

7 Parallels to Dynamics with Outsiders in Botball

Sometimes Botballers need to effectively coordinate with entities who are neither on their team nor on a peer team. Parallels abound here as well.

7.1 Reporting Bugs

It's rare to encounter a high-achieving Botball team that hasn't needed to report a compiler or firmware bug to KIPR at least once. As Botballers are aware, being able to skillfully report a bug greatly improves the chance that it will be fixed promptly. Pertinent skills involve providing accurate and complete steps to reproduce, describing the buggy (and expected) behavior, and identifying any interesting circumstances that the bug might be dependent on.

At Namecoin, we find bugs in third-party software fairly often. Among others, we've found (and reported) security bugs in Brave, Firefox, Chromium, and Windows. Alas, sometimes the quality of the bug report isn't the only factor in getting the bug fixed: while Brave's responses to vulnerability reports has been excellent, we've gotten substantially lower-quality responses from Mozilla, Google, and Microsoft.

Sometimes testing for a bug *once* isn't enough. We're working with Netflix's BetterTLS project [15] and PrivacyTests.org [16] to automatically test major software for TLS-related bugs and proxy leaks, as new versions of that software come out. This has a better chance of surfacing bugs, and also helps average users gauge which software vendors are fixing bugs as they should be.

7.2 Handling Ethics Complaints

Many Botballers have had a negative experience from a team member who can't work with others, behaves irresponsibly, or is otherwise toxic. Sometimes complaints about a team member may come in from external sources as well. Well-run Botball teams need to handle such situations so that one toxic team member can't drag down the whole team, while also recognizing that each case is different, and that in some cases it's not optimal to kick someone out.

At Namecoin, we also occasionally get complaints about team members, and have to determine how to handle them. Between Namecoin's founding in 2011 and the present, three of our team members have been subject to an ethics complaint. In one case, our ensuing investigation yielded a unanimous agreement to expel that team member. In the other two cases, our corresponding investigations determined that the complaints were without merit. (Interestingly, the case where we found cause to expel a team member involved a complaint that was filed by another team member who clearly had an incentive to make sure Namecoin was a friendly community, whereas the two meritless complaints were filed by outsiders with no obvious stake in Namecoin's well-being. I suspect that this correlation is not just a coincidence.) Being able to navigate this kind of situation has been highly important in making sure that Namecoin stays a healthy community.

7.3 T-Shirts and Marketing

On a mildly less mission-critical note, but still important: Botball teams and Namecoin have both derived advantages from clever t-shirts. T-shirts are a highly organic marketing method: if someone sees your Botball t-shirt, and is sufficiently interested or confused about it, they may ask you about it. And that means you're talking to them about Botball. Maybe that conversation will lead them to lobby their school to start a team, or to donate to your team to cover registration fees.

When I was at Norman Advanced, one year our t-shirts stated that we were “Sponsored by American Standard Code for Information Interchange” – a joke reference to our teacher’s name being David **Askey**. Namecoin also uses off-the-wall potentially-confusing jokes on t-shirts as a conversation starter – we have 25 t-shirt designs in production, and routinely add more. Botballers who’ve seen me at GCER will be aware that I wear a different clever t-shirt each day of the conference.

T-shirts are also an effective team cohesion technique. When I was at Norman Advanced, coming up with clever symbolism for our t-shirts was an enjoyable side social activity. Similarly, at Namecoin, we like to bounce clever jokes for t-shirts off each other when we’re not writing code.

8 Conclusion

I’ve documented in previous GCER papers the parallels between Botball and the security/privacy research field in terms of technical skills. But soft skills are important too! And soft skill parallels are commonplace between the scenes as much as technical skill parallels. Kudos are due to KIPR for fostering a useful educational environment for these skills.

And finally, a career plug: if you’re a Botballer, you think privacy should be a human right, and you think the work I’ve described relating to TLS, proxy leaks, or semantic content leaks sounds interesting, send me an email – you might be a good fit for Namecoin. Open source development looks great on resumes or college applications. I can be reached at `jeremy@namecoin.org` or `jeremyrand@danwin1210.de` – send to both addresses. It’s totally fine if you don’t have much experience with these topics – being a Botballer is already a sign that you’re near the top tier in terms of your ability to pick up the material.

9 References

[1] Jeremy Rand. Making HTTPS and Anonymity Networks Slightly More Secure (Or: How I’m Using My Botball Skill Set in the Privacy/Security Field). GCER 2017. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2017/Namecoin_GCER_2017.pdf

[2] Jeremy Rand. From Hacking Botball Controllers to Having My Code in the Number One Privacy Web Browser (Or: What a 2011 Botball Alumni Is Doing in 2022). GCER 2022. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2022/Namecoin_GCER_2022.pdf

[3] Jeremy Rand. I Won Awards for Hacking Botball Controllers; Now I’m Protecting the Anonymity of At-Risk Internet Users (Or: How Botball Prepared Me for a Career in Security/Privacy Research). GCER 2025. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2025/Namecoin_GCER_2025.pdf

[4] Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf

[5] Vincent Durham, et al. Namecoin. https://www.namecoin.org/

[6] The Tor Project. https://www.torproject.org/

[7] Patrick Schleizer, et al. Whonix – Superior Internet Privacy. https://www.whonix.org/

[8] Jeremy Rand. Enhancing Sensor Efficiency with Monte Carlo Subsampling. GCER 2014. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2014/MonteCarlo.pdf

[9] Dan Kaminsky. Black Ops of TCP/IP 2011. https://media.ccc.de/v/cccamp11-4555-black_ops_of_tcpip_2011-en

[10] Jeremy Rand. AR.Pwn: Hacking the Parrot AR.Drone (Part 1). GCER 2013. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2013/AR.PwnPart1.pdf

[11] Jeremy Rand. AR.Pwn: Hacking the Parrot AR.Drone (Part 2). GCER 2013. https://ynqrnzvuotdgvqj2g73f5tyj4rcvwutdemu23jgadawihbkc3lxkbwid.onion/pdf/gcer/2013/AR.PwnPart2.pdf

[12] Alice Margatroid. Leveraging LLMs for Preventing De-anonymization: Occlumask. https://pretalx.riat.at/39c3/talk/UWJ8XB/

[13] Simon Lermen, Daniel Paleka, Joshua Swanson, Michael Aerni, Nicholas Carlini, Florian Tramèr. Large-scale online deanonymization with LLMs. https://arxiv.org/abs/2602.16800

[14] Wikipedia contributors. B. F. Skinner (Reinforcement). https://en.wikipedia.org/w/index.php?title=B._F._Skinner&oldid=1359028279#Reinforcement

[15] Ian Haken. BetterTLS. https://bettertls.com/

[16] Arthur Edelstein. PrivacyTests.org. https://privacytests.org/