NOTES

USENIX Board of Directors

Communicate directly with the USENIX Board of Directors by writing to board@usenix.org.

PRESIDENT
Carolyn Rowland, *National Institute of Standards and Technology*
carolyn@usenix.org

VICE PRESIDENT
Hakim Weatherspoon, *Cornell University*
hakim@usenix.org

SECRETARY
Michael Bailey, *University of Illinois at Urbana-Champaign*
bailey@usenix.org

TREASURER
Kurt Opsahl, *Electronic Frontier Foundation*
kurt@usenix.org

DIRECTORS
Cat Allman, *Google*
cat@usenix.org

Kurt Andersen, *LinkedIn*
kurta@usenix.org

Angela Demke Brown, *University of Toronto*
angela@usenix.org

Amy Rich, *Redox*
arr@usenix.org

EXECUTIVE DIRECTOR
Casey Henderson
casey@usenix.org

Running Virtual PC (vPC) Meetings

Erez Zadok and Ada Gavrilovska
2020 USENIX Annual Technical Conference (USENIX ATC '20) Program Co-Chairs

As the co-chairs of the USENIX ATC '20 PC, our original plans to hold an in-person PC meeting pivoted to virtual PC (vPC) meeting planning due to COVID-19. Along with our very helpful submission chairs (Dongyoon Lee from Stony Brook University and Ketan Bhardwaj from Georgia Institute of Technology), we experimented with three solutions to see what would work best: Webex, BlueJeans, and Zoom. We have now concluded running the vPC meeting, with over 70 participants for at least part of the meeting. Below we describe our experiences in planning and running the vPC.

Ultimately, we settled on Zoom, but it did not solve all of our problems. At this point, we are mainly interested in reporting our experiences while they are still fresh in our memory, in hopes you will find it useful. It would take more time and experimentation to turn this document into a concrete set of recommendations.

Running USENIX ATC is a relatively complex operation for many reasons, including the number of submissions (in the hundreds), and the three tiers of reviewers (numbering almost 120). The two co-chairs and the two submission chairs all need administrative privileges in the online paper reviewing system (HotCRP.com).

vPC Meeting Requirements

1. Our key need for the PC meeting is how to handle conflicts of interest (CoI). In a physical PC meeting, any PC members with a conflict are kicked out of the room, and called back in after the conflicted paper's discussion is over. This requires a waiting room feature.
2. There are numerous tasks that all four of us have to handle efficiently: watching and moving the discussions along, marking decisions, reviewing paper summaries, picking and assigning shepherds, and of course managing conflicts of interest (CoIs). As a result, all four of us need to have admin privileges when running the meeting, not just in HotCRP.
3. We need to verify the identity of PC members, and ensure that only invited individuals can join the meeting after proper authentication.

Webex

Webex allows the host to define alternate hosts. Alas, only one of the alternates at a time can be an active host: once person A delegates host privileges to person B, person A loses host privileges and can't get them back. What we need is true co-hosting, and Webex doesn't seem to support that at the moment.

Webex does have a decent waiting room feature: we were able to manually move attendees to the waiting room and verify that they could not hear or see anything, and could not get back in on their own.

Webex has a very nice registration feature: you invite N people with specific emails and names to a Webex meeting. They are required to register with the email they were invited with, and they cannot change their name.

BlueJeans

BlueJeans supports multiple co-hosts. It also supports a "breakout room," and we were able to move people to it. Alas, people in the breakout room could rejoin the main meeting on their own—clearly undesired. (I guess it's like a conflicted PC meeting member who is outside the main room barging right back in.)

We didn't test BlueJeans's registration feature, as the breakout room problem was a showstopper for us.

Zoom

Zoom has a rudimentary role-based access control system, and allows one host and multiple co-hosts at a time. It allows true co-hosts, which the host can define when creating the meeting, but they need to have a Zoom account. If they don't, the host can easily promote them to co-hosts after the meeting starts. Only hosts can declare others as co-hosts, and the host can even hand off actual host privileges to another co-host, but cannot take them back. All co-hosts have the same admin control over the meeting: they can admit people in/out, un/mute all, etc.

In our experience, Zoom's waiting room worked very well. Participants with conflicts could be kicked out of the meeting and sent into the waiting room, where they could not hear or see anything. We could then re-admit them all with a single click of the admit-all button, and go on to remove the next set of CoI out of the meeting. The key here is that all co-hosts were able to manage these conflicts and the waiting room, allowing us to better parallelize (and double check) this complex task.

Zoom's registration feature is not as good as Webex's. We had to send the Zoom URL to all of our PC members, who then had to register with a valid email and enter their names. They received an email with a personal link to join the meeting—thankfully not a shared URL that could be easily zoombombed. However, they were able to enter any valid email and any first/last name. In theory, someone could create a new dummy email and masquerade as another PC member if they got their hands on the invitation URL.

In the future, we will need to ask PC members to use their proper names and emails that are registered in HotCRP. When the meeting starts, all PC members will be in the waiting room by default, and we'll have to verify one by one whom we are admitting into the meeting—otherwise we can chat privately with them in Zoom to establish their identity. Once we admit everyone, we can turn off the "participants can rename themselves" feature.

Registration becomes even more important for people who will dial in by phone to the meeting. They will still have to register with a per-participant link; then they will receive an email with instructions for connecting to the meeting with a personal phone code that identifies them. When dial-in users connect, they are shown as "Phone User N." We have to identify them by voice and rename them in the Zoom participants list so everyone knows who they are.

Other Solutions?

We heard that at least one PC meeting via Microsoft Teams worked well. Given that we were reasonably pleased with the Zoom setup, and were not sure we had access to test Teams, we did not investigate it. Erez did have the opportunity to join a Teams meeting recently, described below, and we are interested to hear from anyone who has detailed experience with it.

Erez recently joined three different back-to-back meetings with about 6–8 people each, using Microsoft Teams, Zoom, and Google Meet. Overall, he felt that Zoom worked much better and doubted that Teams or Meet would have met our vPC needs.

Microsoft Teams does seem to have a waiting room feature, as Erez had to wait to be admitted, but it's unclear how well it would work for running a vPC. Video and audio quality was lower for some participants; while it might have been their Internet connections, we fear that it might not have scaled to our PC size. Only four people's videos were visible at a time, limiting the ability to feel inclusive and see more people. After examining all the buttons and menu options during the meeting, it seemed to Erez that Teams had far fewer features.

Google Meet also has very few features compared to Zoom, and even fewer than Microsoft Teams. The worst part was that the audio and video quality in Google Meet was considerably poorer for everyone participating. Even turning off everyone's video and streaming audio only, the quality was still fairly choppy.

Webex Teams, which we did not have access to test, reportedly supports multiple concurrent co-hosts.

Experiences from Running the Actual Virtual PC

With a virtual PC, there was more to manage at once. It was important that each organizer use a computer with a large screen—even two screens. We had to have the conference paper management window open as well as the Zoom window, with sub-windows for chat, the participant list, and the waiting room list, our email and messaging client (or cell phone), since people were emailing or texting us with various issues, and a private Slack chat window for the organizers.

When streaming media for hours, some people's computers overheated and shut down after a few hours. It is important to have a sufficiently powerful computer for long-running CPU-hog processes like video and audio streaming.

We used Slack as a side channel for private communications among the meeting organizers. We could have used Zoom's chat feature, but it was too risky—participants could inadvertently broadcast something publicly unintentionally. So we allowed participants to chat only with the host(s) in Zoom. It was useful as people had to tell us about last-minute schedule changes or other requests. The Zoom messaging feature was not very convenient, however, when we needed to send the same message to a few participants (but not all, so as not to violate conflicts), for instance, that their paper would need to be reshuffled in the schedule. Also, Zoom let participants chat with one of the co-hosts but not all of them as a group. Lastly, there was no way to clear the chat history between paper discussions in order to avoid leaking information to other participants once they rejoined.

While Zoom permitted us to manage conflicts as described above, it took time to do so: we had to look up the conflicts in HotCRP, then scroll or search for the right participant in the participant list, then move them to the waiting room one by one. There is no feature for participants to take themselves into the waiting room the way they would during an in-person PC meeting. Zoom, perhaps under network stress, had a delay of 2–3 seconds between when you kicked someone off the meeting and they actually showed up in the waiting room. So it took 1–2 minutes per paper just to manage those conflicts, precious time when you are under a tight schedule. Conversely, in a physical PC meeting, you quickly call the names of all conflicted members, and they all get up at once and leave the room in parallel.

Zoom shows at most 25 participants' video at once, and not all of our participants used their video. (One insisted on calling in from an anonymous phone number due to reports of Zoom privacy concerns.) This made it harder for PC members to know when they could jump in and speak. We tried to manage the order as best we could, calling on people in turn, and we also used the "raise hand" feature a bit, but it still took longer than with an in-person meeting. There were also natural delays in people's audio/video stream and a few people with poor connections. All this added another 1-2 minutes of time when discussing each paper.

When a PC meeting is held in person, people come from all over the world and are present at the start of the meeting at the designated time. But with a virtual PC meeting spanning 12–15 time zones, it was impossible to expect people to be at the meeting at ridiculous early/late hours. So our meeting was scheduled for the middle of the day. We sent a Doodle survey to see what times people could attend, and we tried our best to group papers based on people's time constraints—not an easy task. Worse, because of COVID19, people had day job duties they couldn't ignore, childcare duties, last-minute schedule changes, and more. We had to adapt to people's changing schedules dynamically. This added more "context switching" time between papers.

A few other aspects made the process challenging. First, it was more difficult to control inadvertent leakage of information about paper reviewers—we had cases where either one of us or reviewers themselves asked if we could do paper #X before they left, or when we waited to discuss a paper because of a missing reviewer, but now that information was visible to others—they saw who just joined the meeting. Likely some of this exists in an in-person PC meeting, but probably less so. Second, managing the discussions to wrap up in a fixed amount of time was more difficult, given the lack of other options. PC voting as an option really doesn't work in an online format. We rarely had the full PC, and with people coming and going and videos switched off, it was difficult to tell who was around, whether they would listen in a brief summary of the discussion before voting, etc. As a result, in cases when the PC discussion was "deadlocked" and it was obvious that a reviewer's vote wouldn't resolve it (e.g., an even number of reviewers split 50/50), asking the PC to vote could not resolve the paper's status.

In addition, it was harder to ask the PC members to take the conversation offline and report back—something that's commonly done during in-person PC meetings—because of the above-mentioned issue with time zones and daytime duties. Taking a conversation "offline" meant pushing papers to be decided at some undetermined later point, likely after the actual PC meeting. These two issues made it harder to cut discussions short, which again added to the meeting time.

We already expected that our virtual PC meeting wasn't going to be as effective as an in-person one would have been. So for weeks leading to the meeting, we pushed our PC hard to try and reach a decision on as many papers as possible. That certainly helped a lot (and we have even heard of some PC Chairs who canceled their online PC meeting so they didn't have to deal with the complexities of running it virtually). Still, all these complications caused our PC meeting, originally scheduled for five hours, to take seven hours. And we still had a few of the discussed papers to finalize offline after the meeting.

Finally, a word about security and privacy. Since Zoom saw its user base grow 20-fold in just a few months, it has attracted a lot of media attention and reporting of serious security and privacy concerns. (This is not to suggest that Zoom's competitors' security and privacy practices are perfect and their software bug-free.) As a result, a few high profile communities (e.g., school districts) banned or abandoned Zoom altogether. To their credit, Zoom has apologized publicly, has begun to address these concerns, and has already released several security fixes and new features, promising more. Still, some of our PC members, understandably, preferred not to run the Zoom client or accept their privacy policy as there are reports of numerous Zoom users' credentials sold on the dark web. These users called in via phone instead.

usenix®
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

Writing for *;login:*

We are looking for people with personal experience and expertise who want to share their knowledge by writing. USENIX supports many conferences and workshops, and articles about topics related to any of these subject areas (system administration, programming, SRE, file systems, storage, networking, distributed systems, operating systems, and security) are welcome. We will also publish opinion articles that are relevant to the computer sciences research community, as well as the system adminstrator and SRE communities.

Writing is not easy for most of us. Having your writing rejected, for any reason, is no fun at all. The way to get your articles published in *;login:*, with the least effort on your part and on the part of the staff of *;login:*, is to submit a proposal to login@usenix.org.

PROPOSALS

In the world of publishing, writing a proposal is nothing new. If you plan on writing a book, you need to write one chapter, a proposed table of contents, and the proposal itself and send the package to a book publisher. Writing the entire book first is asking for rejection, unless you are a well-known, popular writer.

;login: proposals are not like paper submission abstracts. We are not asking you to write a draft of the article as the proposal, but instead to describe the article you wish to write. There are some elements that you will want to include in any proposal:

- What's the topic of the article?
- What type of article is it (case study, tutorial, editorial, article based on published paper, etc.)?
- Who is the intended audience (syadmins, programmers, security wonks, network admins, etc.)?
- Why does this article need to be read?
- What, if any, non-text elements (illustrations, code, diagrams, etc.) will be included?
- What is the approximate length of the article?

Start out by answering each of those six questions. In answering the question about length, the limit for articles is about 3,000 words, and we avoid publishing articles longer than six pages. We suggest that you try to keep your article between two and five pages, as this matches the attention span of many people.

The answer to the question about why the article needs to be read is the place to wax enthusiastic. We do not want marketing, but your most eloquent explanation of why this article is important to the readership of *;login:*, which is also the membership of USENIX.

UNACCEPTABLE ARTICLES

;login: will not publish certain articles. These include but are not limited to:

- Previously published articles. A piece that has appeared on your own Web server but has not been posted to USENET or slashdot is not considered to have been published.
- Marketing pieces of any type. We don't accept articles about products. "Marketing" does not include being enthusiastic about a new tool or software that you can download for free, and you are encouraged to write case studies of hardware or software that you helped install and configure, as long as you are not affiliated with or paid by the company you are writing about.
- Personal attacks

FORMAT

The initial reading of your article will be done by people using UNIX systems. Later phases involve Macs, but please send us text/plain formatted documents for the proposal. Send proposals to login@usenix.org.

The final version can be text/plain, text/html, text/markdown, LaTeX, or Microsoft Word/Libre Office. Illustrations should be PDF or EPS if possible. Vector formats (TIFF, PNG, or JPG) are also acceptable, and should be a minimum of 1,200 pixels wide.

DEADLINES

For our publishing deadlines, including the time you can expect to be asked to read proofs of your article, see the online schedule at www.usenix.org/publications/login/publication_schedule.

COPYRIGHT

You own the copyright to your work and grant USENIX first publication rights. USENIX owns the copyright on the collection that is each issue of *;login:*. You have control over who may reprint your text; financial negotiations are a private matter between you and any reprinter.