
# copilot

**DECEMBER 1, 2014 - JUNE 30, 2015**

**SUBMITTED TO:**
The Knight Foundation
Knight Prototype Fund Grant

**PROJECT CONTACTS:**
Seamus Tuohy, Sr. Technologist and Risk Advisor
Email: stuohy@internews.org

Megan DeBlois, Program Coordinator
Email: mdeblois@internews.org

# Table of Contents

# 1. ACKNOWLEDGEMENTS

The Co-Pilot project was made possible through funding from the Knight Foundation, and continuous support and cooperation of Chris Barr and Nina Zenni.

Co-Pilot is a product of Internews' Internet and Communications Technology (ICT) program. Co-Pilot was designed by Seamus Tuohy (Sr. Technologist and Risk Advisor, ICT Programs) and Megan DeBlois (Program Coordinator, ICT Training Programs). Research implementation was led by Megan DeBlois. Technical development was led by Seamus Tuohy.

We are indebted to the trainers and developers who volunteered their deep knowledge and critical feedback during the research phases of the project. For privacy reasons, we will not list all of your here. A special thanks to Internews' Nick Sera-Leyva for his insights as a trainer and his support with the trainer community

We would also like to acknowledge the Internews management team, and specifically Jon Camfield for his technical review and inputs during the peer review processes, and his deep commitment during the development phase of the project.

## 2. ABOUT INTERNEWS

Internews is an international nonprofit organization whose mission is to empower local media worldwide to give people the news and information they need, the ability to connect and the means to make their voices heard. Formed in 1982, Internews has worked in more than 90 countries, and currently has offices in Africa, Asia, Europe, the Middle East, Latin America and North America. With global expertise and reach, Internews trains media professionals and civil society, introduces innovative media solutions, increases coverage of vital issues and helps establish policies needed for open access to information.

Internews is a pioneer in developing and enhancing information security tools and practices, increasing the use of circumvention and mobile technologies, and improving secure communication channels to ensure journalists, civil society and human rights defenders have access to independent information. Internews has a robust physical and digital security program with staff who provide extensive support on-the-ground and remotely. The program has managed the development, deployment, and scaling of several new circumvention tools (Tor, Psiphon, Lantern), secure mobile technologies (Orweb, Orbot, ChatSecure), and a wealth of resources (LevelUp, SaferJourno, and SAFETAG) to enable and empower individuals and organizations to better identify and resolve their digital vulnerabilities more effectively.

# 3. EXECUTIVE SUMMARY

Over the first six months of 2015, Internews' ICT programs, with support from a Knight Prototype Fund Grant, developed Co-Pilot; an open source tool that enables a digital security trainer to simulate hostile digital environments.

**Problem:** The production of high-quality, effective materials for digital security trainings require significant insight, craftsmanship, testing, iteration, and — crucial to its adoption - support and buy-in from the wider training community. Experienced digital security trainers are very discerning about the materials they use to supplement their trainings, and their acceptance of materials will in turn set norms for novice trainers as they join the community.

**Objective:** Taking this into consideration, the Co-Pilot team adopted a human-centered design approach to develop this tool, engaging digital security trainers in identifying the requirements and barriers that Co-Pilot needed to address in order to be adopted in this community. The outcome of this process was to develop a cross platform tool for embedded devices that would act as both a wireless hotspot for participants as well as a censorship system.

This process resulted in the successful development of the Co-Pilot tool, which has the following features:

- A wireless hotspot that allows trainers to provide a safe environment for participants to explore censorship.
- Easy-to-use interfaces that allows trainers to have fine-grained control of censorship environments.
- Pre-loaded DNS censorship plugins for DNS Blocking and Redirection.
- A plugin system that allows developers to easily add new censorship and surveillance functionality.
- Support for three embedded hardware platforms for under $100 USD.

This report summarizes the major findings, limitations, and future recommendations that were derived from the human-centered research & design process of Co-Pilot.

### 3.1.1. Findings

- **Adoption:** Making Co-Pilot valuable to the trainer community
- **Adaption:** Ensuring the growth and long-term stability of Co-Pilot

### 3.1.2. Limitations

- **Duty of Care:** Simulating hostile environments without causing trauma

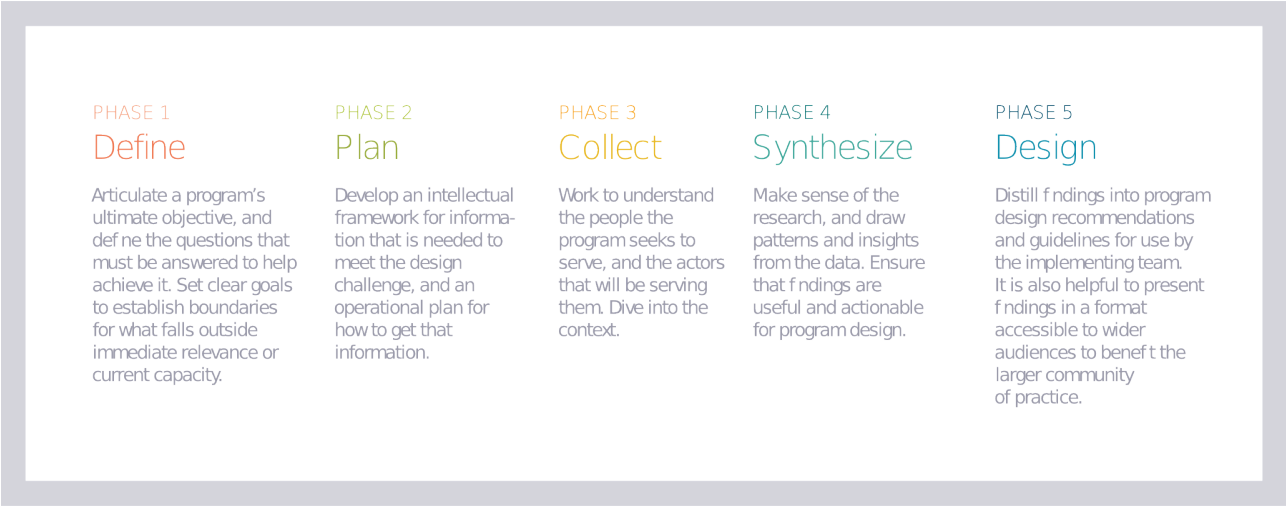
### 3.1.3. Recommendations

- **Simulations:** The creation and sharing of accurate reproductions of regional censorship
- **Context Appropriate:** Supporting the diversity of training environments
- **Documentation:** Removing trainer uncertainty

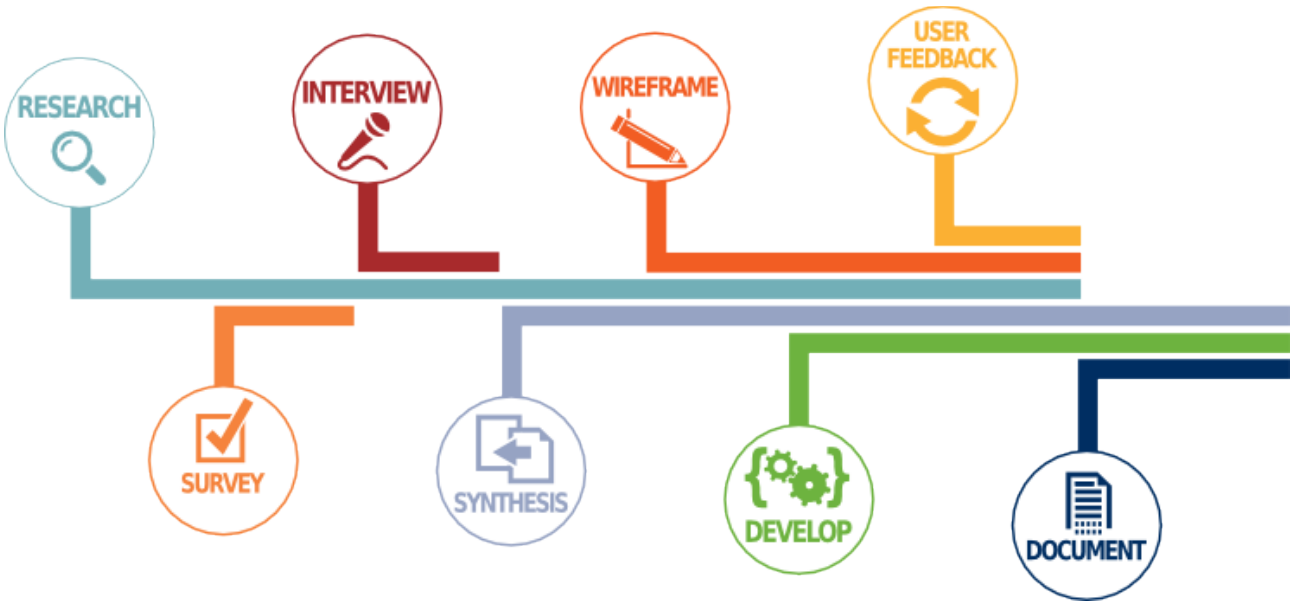
# 4. RESEARCH APPROACH

The Co-Pilot team determined that an initial needs assessment was a necessary first step in order to ensure that the Co-Pilot prototype addressed the core issues digital security trainers face in the field. The main goals of the initial needs assessment were two-fold:

- to better understand the current environment trainers work in; and
- to collect and receive feedback from trainers regarding what they would like to see in a training aid (such as Co-Pilot.)




Internews has developed its own research design methodology through its experience in media development in partnership with Reboot, conducting research in different types of information and media ecosystems in Pakistan. The project produced a highly useful practitioner's guide to research, laying out a clear and cohesive methodology and approach. The team repurposed this comprehensive methodology, and applied it as the development and design framework for Co-Pilot. To ensure that Co-Pilot meet the needs of the digital security trainer community, the first task that the Co-Pilot team took on was to conduct research to better understand the current training environment, trainer needs, what caused trainers to adopt or reject new methods, materials, and tools, and to identify high-need topics to guide Co-Pilot development. The core of this work was done through desk research, surveys, interviews, and user testing using wireframes while informal research continued throughout the entire process. The team's deep networks in the digital security community were very useful in having the availability and willing participation of a trusted network of experienced digital security trainers to complete surveys, participate in interviews, and explore the interface. The Co-Pilot team's attendance at trainer and developer events also allowed for invaluable face-to-face access to parts of the community around the world that the team could not otherwise afford to meet with in person.




The Co-Pilot team conducted desk research on the technical feasibility and exiting training tool landscape. This desk research continued throughout all research phases to supplement information gained in other phases.

The Co-Pilot team conducted an initial needs assessment survey to ensure the prototype addresses the core issues digital security trainers face in the field. The initial survey was sent out to 30 members of the digital security community, with the vast majority training regularly, and received 20 responses. Trainer selection was largely done based on the trainer's experience in the space and knowledge on the topics.

Out of the 20 survey responses received, all agreed to follow-up interviews. The team conducted 4 in-depth participatory ideation interviews with technologists and trainers that had identified either unique possible use-cases or key barriers to trainer adoption during the survey phase.

The team analyzed the survey and interview responses to identify the initial set of features for Co-Pilot and identify additional research that would need to be conducted by the team.

The team conducted think aloud testing with five trainers where they asked the trainers to speak their thoughts as they moved through the mock wireframes of the user interface. This process allowed the team to understand how trainers wanted to interact with Co-Pilot and where the user-interface needed to be modified to support their needs.

With feedback from the wireframes the Co-Pilot team began technical development of Co-Pilot. Technical development was done publicly on Github where the rapid and flexible development process was connected to the research process through a public issue queue.

Throughout the development process the Co-Pilot team conducted formal and informal demonstrations to trainers and in-depth conversations with developers. These activities helped the team continuously evaluate their prioritization of findings from the original research phase.

Once the Co-Pilot device was completed the Co-Pilot team focused on documenting Co-Pilot's setup, use, and modification, The team also took time to document their research findings to support future development in this area of work (this document).

# 5. MAJOR FINDINGS

## 5.1. Adoption: Making Co-Pilot valuable to the trainer community

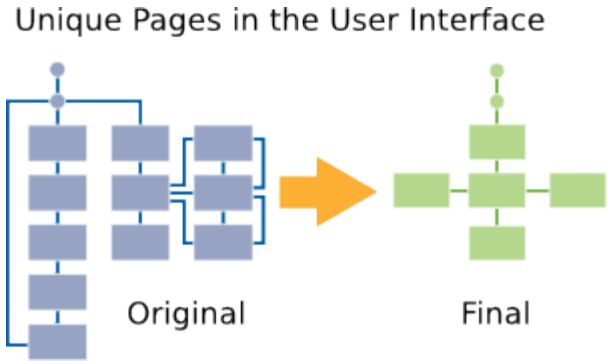
When asked what would cause a trainer to use or not use a tool like Co-Pilot, the majority of interviewees wrote that ease of setup and use was critical. While not entirely unexpected, this was a major theme in all questions that explored adoption by trainers. Understanding that ease of use was one of the most important features of Co-Pilot, multiple iterations of usability testing occurred throughout the development process – with both trainers and training participants.


Initial Wireframes | Version 1 Prototype | Final Design

Many digital security trainers come from non-technical backgrounds where they were required to learn defacto digital security out of necessity. One of the challenges they face is that security tool makers often design their tools for those with a high level of technical expertise, and not for their mainly non-technical expert users. It was clear from the team's research that Co-Pilot needed to be intuitive for both the tech-savvy trainer and the trainer that does not have a strong technical background.

> "For junior trainers if the tool is too hard to set up or the environments (and the differences between them) are too difficult to understand and communicate it will limit uptake." - Interviewee

When conducting user-interface testing the team identified "interface fatigue" as one of the key barriers for unfamiliar users to feel comfortable with Co-Pilot. New user interface elements forced users to learn new ways of interacting with Co-Pilot. Co-Pilot started out with a variety of different menu types and interactive elements on every page. Based upon this feedback the Co-Pilot team identified a minimal number of user interfaces required to meet the functionality requirements of Co-Pilot and repeated interactive elements across pages as much as possible.



"I'd want to be able to switch very easily between environments for two situations… Anyway, I should probably put the emphasis on switching easily - I just want to push a button and – voila! – the change happens." - Initial Survey

"Training day you would turn it on and be able to transition to multiple blocking configurations pretty seamlessly" - Interviewee

Trainers were clear that Co-Pilot could not slow down the pace of their training by requiring lengthy configuration when they wish to switch from one censorship environment to another. In fact, any implementation of Co-Pilot that forced trainers to modify their existing training methods in any significant way would have little to no adoption within the training community. The team responded to this feedback in the user interfaces by allowing rapid creation, modification, saving, and loading of profiles. By combining options (such as save & apply), and decoupling back-end systems to allow for shorter profile reload time, the Co-Pilot system can switch censorship profiles in a manner of seconds.

Experienced digital security trainers have spent years refining their training session content, pedagogical approaches, and support materials to increase their training's effectiveness and improve behavior change in their participants. When trainers are crafting their agenda they weigh each component of the training against the needs and goals of the trainees. For Co-Pilot to gain adoption, the feedback demonstrated a need for the tool to be able to supplement trainers existing curricula, not divert it. The Co-Pilot team responded by creating a sample training module that uses Co-Pilot to support an existing activity from the LevelUp training community. Through this the Co-Pilot team hopes to expose how easily Co-Pilot can be integrated into trainers existing practice.

"Really it lies with the responsibility of the trainer to stay up to date on the happenings of the specific countries/regions they work in and bring this knowledge and mitigation techniques to combat these specific cases to the training." - Interviewee

From the onset of the project the Co-Pilot team decided that it was out of the scope of the project to develop stock profiles that could meet the multitude of training curricula developed in the highly ad-hoc digital security training sector; nor did they believe they could maintain profiles to keep up to date with curricula as it changes to keep up with increasingly closing environments worldwide and rapidly updating tools. As such, trainers need to be able to modify Co-Pilot to provide timely censorship simulation environments that fit their unique needs without technical support. The Co-Pilot team chose to abstract the censorship environment creation process into a easy to use "censorship profile" interface that allows trainers to have intricate control over the censorship Co-Pilot implements without forcing a deep understanding of the technology.

"The most important feature would be for the tool to be able to block access to websites that trainees use a lot like Gmail, Yahoo, Facebook and Twitter during the training." - Initial Survey

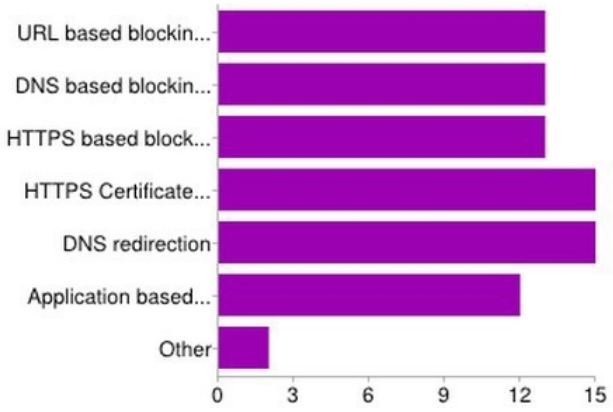
Tool-focused trainers needed a tool that would fit into their existing tool focused trainings, and not force them to add extra content where they were forced to explain how censorship and circumvention worked. This sub-set of trainers was large enough to lead the team to remove all features related to educational visualization or educational content from the initial Co-Pilot roadmap. By focusing on features that supported the full range of training types, Co-Pilot's team hopes to be able to provide the greatest degree of support to the entire digital security training community.

The underlying functionality expressed by all trainers was a need for participants to verify that the tool they were being trained on worked, and that they were using it correctly. For civil society groups, activists, and human rights actors, digital security tools are a burden to their work. If they use them incorrectly, it can put their life at risk, but time spent with these tools is time they cannot spend actually doing their work. If members of these groups are unsure about the efficacy of a tool or their ability to use it correctly, they will stop using that tool or use it incorrectly and put themselves at risk. Co-Pilot supports trainers in providing experience-based decisions on when using a tool is appropriate, and allows participants to confirm if they are using that tool correctly. This is highlighted in Co-Pilot's sample training activity. This activity contains both the traditional method of teaching participants how to manually verify if the circumvention tool being trained on is working, as well as an ongoing Co-Pilot demonstration which will block parts of the Internet until the participants have successfully used the tool to circumvent its local censorship.

> One interviewed trainer's personal goal was to have "participants doing their own thing independently so they could be checking if they have set up the application successfully on something like Co-Pilot at their own pace."

When asked which types of hostile digital environments would be most beneficial to simulate the highest response from trainers was a request for HTTPS certificate spoofing and DNS redirection – both active manipulation of the network. Passive blocking/throttling methods closely followed. Trainers were clear in their desire to train on "visible" examples of censorship. These types of censorship have visual indicators that allow trainee's to clearly identify that censorship is taking place, while in other censorship environments, a trainee cannot be certain if they are experiencing active censorship or connectivity problems.

**What topics would benefit the most from an environment like this?**




| | | |
|---|---|---|
| URL based blocking / throttling | 13 | 65% |
| DNS based blocking / throttling | 13 | 65% |
| HTTPS based blocking / throttling | 13 | 65% |
| HTTPS Certificate Spoofing | 15 | 75% |
| DNS redirection | 15 | 75% |
| Application based blocking / throttling | 12 | 60% |
| Other | 2 | 10% |

> "I'd love it if there was a DNS redirection feature. This is something so fundamental to the Internet that we don't normally pay attention to it." - Initial Survey

The Co-Pilot team focused on implementing the “Domain name system (DNS) blocking and redirection” capabilities during the prototype phase. These types of censorship allow a trainer to specify a number of individual URL's (www.google.com) or a range of URL's (for instance, anything ending in .com), and Co-Pilot will subsequently intercept all participant requests for those URL's. If the trainer chooses to **block** a URL, the participant will be unable to find the location of those websites on the Internet. On the other hand, if the trainer choose to **redirect** a URL, the participant's browser will take them to a webpage that lives on Co-Pilot, which displays random images of graffiti from the March 2014 censorship of Twitter in Turkey instead of their desired

website.[1]

> "No projects are ever built with future support." - Trainer Interviewee

From the initial project design, the team was aware of the importance for an open-source project to create an environment that would encourage outside contribution, and adaptation by the community. The digital security tool space is littered with examples of abandoned and unfunded projects that trainers and developers alike have previously praised as vital contributions to the space. This has made developers and trainers wary of new tools, and outright avoid projects that do not have avenues where they can engage. Co-Pilot aimed to support all of these needs by actively using its GitHub issue queue for most project communication during the brainstorming and development phases, the project wiki for documentation throughout the project, and by maintaining a clear roadmap of project goals that documented milestones for funded development and research based future milestones to guide thei contributions of other developers.

Throughout the Co-Pilot research and development phases the team was approached by trainers and developers many times about running Co-Pilot on the Raspberry-Pi device. Among embedded devices the Raspberry-Pi has one of the greatest levels of brand awareness, and there are many people who are interested in finding a reason to purchase one. Even when the many limitations of the device were discussed, the interest in this device rarely wavered. Eventual support for the Raspberry-Pi pushed the team to add support for a variety of platforms to their roadmap. The team eventually had to drop possible support for the original Raspberry-Pi because the platform did not have the capabilities needed to allow trainers to easily install and run Co-Pilot. While the team initially wished to dismiss the Raspberry-Pi, the team quickly realized that support for this high-visibility platform would greatly increase adoption and the long-term stability of the project and prioritized Raspberry-Pi 2 support. The Raspberry-Pi 2 had the proper level of hardware support along with the high-value brand identity that made Raspberry-Pi support vital.

Sample Question: What would prevent you from using an environment like this for your trainings?

- "A proprietary device."
- "Form factor is an interesting issue. If it fits in the pocket, it's a no-brainer, if it's a big ol' Linksys cigar box, can't always roll with that on board."
- "It would be nice to be able to run this off of a Raspberry Pi or another sub-100 USD [system on a chip]."

Adding support for the variety of devices that developers and trainers desired pushed the Co-Pilot team to seek out a stable core platform that worked across a variety of requested devices. The Co-Pilot team identified the Kali Linux penetration testing system as the perfect platform. This platform has a set of build scripts that allow for stable installation on on a variety of embedded hardware devices. Co-Pilot currently runs on three embedded hardware devices and has possible integration across an additional five devices because of its Kali-Linux base. The Kali-Linux distribution also has built within it a variety of specialized tools for traffic sniffing (surveillance), spoofing, and wireless attacks that developers can use to easily extend co-pilots capabilities. By using these well known, stable, open-source platforms and tools Co-Pilot's team could extend its supported devices while adding high-impact censorship and surveillance simulations easier for

1 Graffiti was used during the March 2014 censorship to share address' of DNS servers which would, unlike the countries DNS servers, provide the correct address for Twitter.

outside developers.

## 5.2. Adaptation: Ensuring the growth and long-term stability of Co-Pilot

"The next time I go to [country X] I would love to be able to give [trainers there] a [Co-Pilot] config file saying this is what I used a couple months ago [last time] I was there" - Interviewee

One of the benefits of having a wide range of digital security trainers take the survey is being able to see its applicability in a variety of different settings and contexts around the globe. Understanding the value that trainers from different regions and contexts see in this type of tool also has implications for the tool's design. Many trainers mentioned specific countries when talking about demonstrating censorship environments. This led to internal discussions on how to incorporate country-specific censorship and blocking into the Co-Pilot prototype. Given that country-specific censorship is difficult to track and takes a significant amount of resources to stay up-to-date, the Co-Pilot team focused instead on the ability to share different blocking configuration profiles with other trainers.

"I am sure folks would participate and help contribute content for different censorship environments" - Interviewee

Profile sharing allows trainers to pool resources, while also allowing them to leverage existing digital security and tech community research that is occurring throughout the digital security community. With the varying level of expertise among trainers, the ability to share profiles allows technical trainers to provide tailored profiles that accurately highlight the capabilities of various digital security tools, while trainers with greater facilitation experience can share profiles that are crafted to support a specific trainings mode of delivery, cultural considerations, and pedagogical goals. Therefore, the team implemented the ability to download and upload profiles from the interface directly to their personal device.

It is known in the open-source community that if an open-source project wishes to gather outside contributions and adaptations, it needs to also make it easy for developers to engage with, and contribute to the project. But surprisingly, many projects require a developer to read through list-serv's and issue queues, beg for advice in chatrooms, and read pages of source code in order to figure out how to add even the smallest feature to a tool. These developers will seek out alternative uses for their free time if a project has not been built for modification and optimized for their success. The team faced continuous requests from developers who wished to add different types of functionality to Co-Pilot. Just as the ability to create tailored profiles was critical to the long-term adoption and use of Co-Pilot, developers saw the ability for outsiders to extend the Co-Pilot software as vital to its long-term viability as a core digital security training tool. In response to this Co-Pilot was re-engineered to include a plugin system at its core. This plugin system allows for Co-Pilot's capabilities to be extended without having in-depth knowledge of how Co-Pilot's back-end works or having had experience working with its underlying code-base. In order to support developers using the plugin system the Co-Pilot team also created an example plugin, documentation for creating and installing a plugin, and a plugin management menu that allows developers to monitor and restart a plugins from the Co-Pilot web interface.

# 6. LIMITATIONS

## 6.1. Duty of Care: Simulating hostile environments without causing trauma

There was a split within the trainer community between those who requested that Co-Pilot implement surveillance, faux malware delivery, and other active digital attacks and those who felt that implementing these could end up harming those we wish to help. The practice of **controlled exploitation** is used by some trainers to increase the sense of urgency or vulnerability of participants. Examples of controlled exploitation can include a trainer changing a participants background when they leave their computer logged in while out of the room, showing participant's user-names and passwords that are exposed in wireless traffic during the training, or even using a wireless device, like Co-Pilot, to replace all images on websites that participants browse during the training with one of the trainers choosing.

> "This can often scare people and we know that when people are scared, they are not in a good state of learning; in fact sometimes they will start to shut down and not take any information in and then you are wasting time, money, resources, etc." - Interviewee

> "I'd like to be able to send my trainees a spear-phishing e-mail with an attachment which successfully infects them because their software's not patched." - Initial Survey

> "I don't want to encourage trainers to abuse this feature" - Interviewee

Experienced trainers, like those who requested these features, have experience balancing the psychosocial needs of their participants during training's. It takes considerable experience and skill to actively attack participants in a training without causing alienation. The team wanted Co-Pilot to be able to be used by both experienced and junior trainers. Without proper guidance on managing psychosocial care when actively attacking participants Co-Pilot could harm the effectiveness of junior digital security trainers.

With the limited time-line for core feature development, the Co-Pilot team decided to focus on only censorship features. As attack style features are explored in the future the team will reach out to the trainer community to identify how to provide proper disclaimers and training support to trainers wishing to incorporate them into their trainings.

> "It needs to be created in a form that doesn't allow the tool to be downloaded and used nefariously. Like, let's not inspire trainers who are also parents to start censoring their kids' Internet use." - Initial Survey

The team cannot prevent Co-Pilot from being used maliciously. One cannot create software that leverages malicious technology without accepting some level of risk that others will use their software in an unintended manner. Knowing this, the team, therefore, decided that they would target the motivation behind most software reuse; commercial gain. The team applied an open-source and non-commercial license to the software so that Co-Pilot could not be merged into commercial or proprietary small-scale censorship projects. This is a minor mitigation for a risk that the team almost entirely had to accept in order to gain the adoption, use, and adaptation that would be needed to make an impact in the trainer community.

# 7. RECOMMENDATIONS

## 7.1. Simulations: The creation and sharing of accurate reproductions of regional censorship

Early in the project there was a misconception that Co-Pilot was attempting to provide accurate simulations of the censorship being used in various countries. This misconception led to a flood of initial feedback on the value that this would provide in advocacy, education and circumvention tool development. This feedback made it clear that the ability to incorporate country-specific censorship and blocking into Co-Pilot could be valuable.

> "The ability to turn on and off specific types of censorship in a very granular way – AND have a list of notable censorship regimes that are known to rely on any particular technique (so as to make it clear that the trainee's experiences in the training session are actually very realistic and applicable to real-word environments)." - Initial Survey

Supporting these simulations, while possibly quite valuable, have challenges that led the team to keep them off the current implementation roadmap. Country-specific censorship is difficult to track and keeping accurate, and up to date analysis takes a significant amount of resources. Accurate simulations add realism to the training environment at the expense of trainer control. When trainers use the current Co-Pilot they have full control of the censorship that the participants will encounter. Trainers can craft censorship profiles that specifically support the lessons that they are teaching. By using accurate simulations of a country the trainer would lose the fine grained control over the Co-Pilot environment that they need to directly support their individual lessons. Furthermore, the possibility of a trainer experiencing unexpected censorship that they cannot diagnose as censorship or connectivity issues adds a technical barrier to use that subverts the basic Co-Pilot goal of supporting non-technical trainers.

## 7.2. Context Appropriate: Supporting the diversity of training environments

Digital security trainers face a range of challenges in their work environment. Inconsistency in the quality of venues, access to training materials, and Internet bandwidth in a training space are just a few of the challenges that trainers face. In the future, Co-Pilot can be extended to respond to, and even address, some of these challenges.

> “Usually participants will get frustrated if they can’t download [software or training materials] right away and then will start passing USB sticks around, which is the worst!” - Interviewee

During the interview process trainers were most concerned about how to use Co-Pilot when there was only wireless connectivity available. Currently the easiest way to accomplish this task is to use a portable hotspot to provide a bridge between the available wireless and the Co-Pilot device. The Co-Pilot team discovered this when there was only wireless connectivity during one of their demonstrations. Technically, Co-Pilot is able to create a bridge between any available WiFi hotspot and the network it provides without external hardware support. This currently requires a trainer to use Co-Pilots text-only command line interface, which requires a level of technical expertise that puts it out of the hands of most of the trainer community. In the future the

team would like to add this functionality to the Co-Pilot web interface. This would allow trainers to use Co-Pilot in a greater range of venues.

Especially slow Internet can cripple a training if a trainer has participants accessing training resources and software from the Internet. More troubling is when access to the Internet, or even power, is entirely unavailable for large periods of time. These situations render Co-Pilot's censorship capabilities useless, as they require access. But, the Co-Pilot device can still provide local connectivity that could allow a trainer to take a training offline. Traditionally, trainers keep copies of their training materials and software on USB, CD devices in case they are inaccessible during a training. Recently, some trainers have begun using portable tools like LibraryBox that can distribute training materials (presentations, guides, video-clips, how-to's) and software locally over a wireless link to make this process easier, and far less likely to pass malware between computers. Adding this functionality to Co-Pilot will make it a valuable addition to a digital security trainers tool-kit no matter what connectivity challenges they face.

## 7.3. Documentation: Removing trainer uncertainty

In the findings, limitations, and recommendations above a core theme is apparent - Co-Pilot can be a valuable and long-term addition to the trainer community if its possibilities and proper use are understood. The team has already created base documentation for Co-Pilot, but as new features are added, and as the community who uses Co-Pilot widens to those who do not have some level of connection to the Co-Pilot team ongoing updates to documentation will be needed to respond. Opportunities for documentation are evident to:

- explain how to integrate Co-Pilot into existing trainings would be a valuable aid to adoption by the expert community;
- explain networking concepts that are used in Co-Pilot censorship would allow trainers to answer participant questions they encounter; and
- describe how to run Co-Pilot using a wireless access point, or off of an external battery pack would increase its utility in a variety of environments.

Beyond this, greater integration of Co-Pilot documentation into the interface itself, through more robust and interactive help text, would allow trainers to get access to documentation that they need without having to seek it out on the Internet.

# 8. CONCLUSION

The digital security training community is a loosely knit web of individuals from across the globe. These individuals work in a range of hostile environments, and have an even wider range of experience and expertise. The Co-Pilot team applied a human-centered methodology to this community in order to develop a tool that they would not only want to use in their trainings, but a tool that they would want to adapt and build upon in the future. The feedback collected from the training community during this process had a deep impact on the design, development, and future path of the Co-Pilot project. The result of this six month project is a tool that meets the functional needs of the digital security training community and is tied deeply into their existing practices.