BeyondCorp 5
The User Experience

VICTOR ESCOBEDO, BETSY BEYER, MAX SALTONSTALL, AND FILIP ŻYŹNIEWSKI

Victor Escobedo is a Corporate Operations Engineer at Google in Mountain View. Originally joining Google in 2010 through the ITRP Program, he now focuses on change and impact management. He holds a BS in computer science from CSU Fullerton. victore@google.com

Betsy Beyer is a Technical Writer for Google Site Reliability Engineering in NYC. She has previously provided documentation for Google Data Center and Hardware Operations teams. Before moving to New York, Betsy was a lecturer in technical writing at Stanford University. She holds degrees from Stanford and Tulane. bbeyer@google.com

Max Saltonstall is a Technical Director in the Google Cloud Office of the CTO in New York. Since joining Google in 2011, he has worked on video products, internal change management, IT externalization, and coding puzzles. He has a degree in computer science and psychology from Yale. maxsaltonstall@google.com

Filip Żyźniewski is a Site Reliability Engineer at Google in Dublin and the lead of BeyondCorp's portal project. He previously worked as a Performance Engineer at Sabre Holdings. He holds a master's degree in computer science from the University of Lodz. zyzniewski@google.com

Previous articles in the BeyondCorp series discuss aspects of the technical challenges we solved along the way [1–3]. Beyond its purely technical features, the migration also had a human element: it was vital to keep our users constantly in mind throughout this process. Our goal was to keep the end user experience as seamless as possible. When things did go wrong, we wanted users to know exactly how to proceed and where to go for help. This article describes the experience of Google employees as they work within the BeyondCorp model, from onboarding new employees and setting up new devices, to what happens when users run into issues.

Enabling a Seamless New Hire Experience

For many new employees, the idea of a BeyondCorp model is quite foreign: they're used to accessing the tools they need for their day-to-day work through VPNs, "corp wireless," and other privileged environments. When we initially rolled out BeyondCorp, many new hires continued to request VPN access from our help desk team (internally known as Techstop). From past experiences, they assumed they needed to jump through a few IT hoops if they planned to work while away from the office. The architects of BeyondCorp mistakenly assumed that users would try to access internal resources while away from the office and notice that things "just worked"—no access requests from users and no support load for Techstop would be a win-win!—but old habits die hard.

New Hire Orientation

We clearly needed to reach users earlier in their IT journey at Google, so we began introducing BeyondCorp in new hire orientation. During orientation, we explicitly avoid explaining the technical aspects of the model and instead focus on the end user experience. We emphasize that users don't need VPNs and that they're "automatically" granted remote access; they can work from the office, from their home, on a plane, or in a coffee shop without changing their workflows. During this short training, we show users the BeyondCorp Chrome extension—the most common user-facing expression of the BeyondCorp access model (for more details on the extension, see "The BeyondCorp Extension," below)—and the icon that represents a "good" connection within BeyondCorp (see Figure 2). We explain that from a good connection, they can access the vast majority of the tools and resources they need from any network connection.

New Device Setup

When users log in to their corporate devices with their corporate credentials the first time, their access settings are automatically configured. To enable this seamless onboarding experience, inventory processes and platform management tools work behind the scenes to configure a new hire device for initial setup. As described in [1], we infer device trust based on a number of signals, some observed (last security scan, patch level, installed software, etc.) and some prescribed (assigned owner, VLAN, etc.). To handle this complexity, our inventory teams follow an automated provisioning process to ensure that new hire devices are correctly trusted at first login. Once the necessary user credentials are validated,

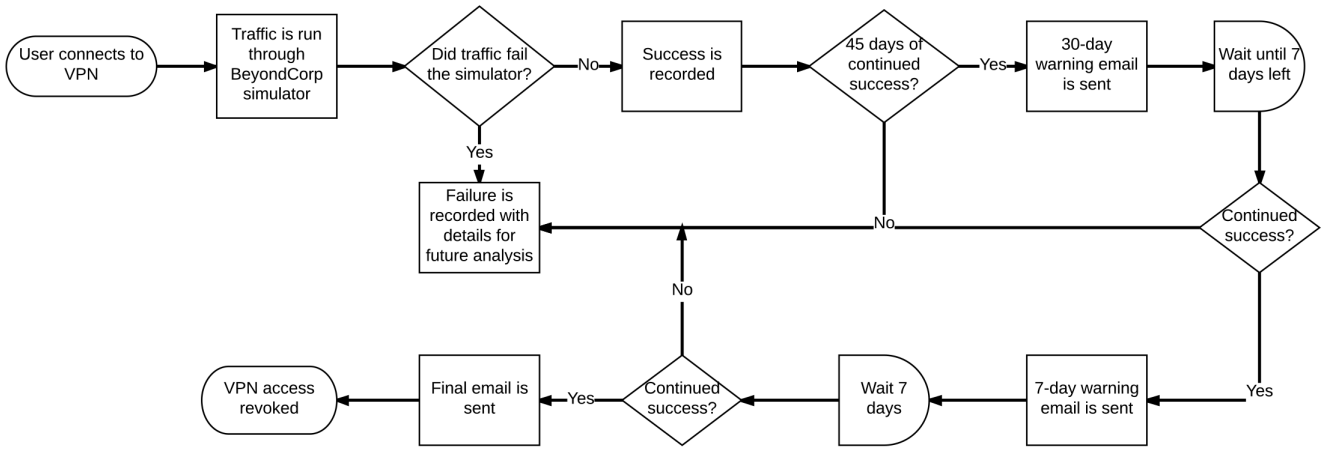

Figure 1: Automated analysis and revocation of employee VPN usage

we automatically push our custom Chrome extension to the machine.

From the user's perspective, as long as they see the green icon in the extension, they know they can access their corporate resources. By explaining the BeyondCorp Chrome extension in new hire training, we have virtually eliminated new hire confusion and support requests relating to remote access.

VPN Reduction

Although new hires learn about BeyondCorp during orientation, their first few days at Google can be a somewhat overwhelming torrent of information. Because we don't expect every person to recall every detail they learn that first day, we modified our VPN request processes and tools to emphasize the concepts introduced in orientation.

Since new hires aren't given access to our VPN gateways by default, they must request VPN access through an online request portal. On this portal, we clearly remind users that BeyondCorp is automatically configured and that they should try to access the resources they need before requesting VPN access.

As shown in the flowchart in Figure 1, if the user skips this warning, we also perform automated analysis on the services users access through the VPN tunnel. If a user hasn't accessed a single corporate service not available within the BeyondCorp model within 45 days, we send them an email. The email explains that because all the corporate resources they've accessed are supported through BeyondCorp, their VPN access will expire in 30 days unless they access a service that isn't supported by BeyondCorp. We send one more notification seven days before their VPN access expires, and then revoke permission to the VPN gateway at the end of the seventh day. This automated process allows us to proactively cull unnecessary usage of legacy access infrastructure, and will eventually allow us to turn down our VPN infrastructure entirely.

Loaners

As a side benefit of the automatic configuration implemented for BeyondCorp, we've also improved other technology experiences for our users. One of the most visible improvements is our loaner laptop program. Like many modern companies, our employees are quite mobile and freely work from their desks, meeting rooms, lounges, or their homes. Mobile devices—specifically, laptops—are incredibly vital to their productivity. To handle cases of forgotten, misplaced, or stolen laptops, we have a self-service loaner laptop program that gets users up and running again as soon as possible.

Using custom-built Chromebook loaner stations deployed around the world, any user can temporarily assign a loaner laptop to themselves for a period of up to five days. Users benefit from the ability to simply pick up a laptop and get back to work within a matter of minutes. Techstop benefits from fewer requests for loaners, which frees up their time to work on other issues. When the user returns the device or the loaner period expires, the system automatically revokes the certificate and demotes the device's trust, leaving it ready for the next user to reinitiate the loaner process.

The BeyondCorp Extension

By more or less eliminating the need for a VPN client, we can encapsulate almost all access needs—whether remote or onsite—through one entry point, the BeyondCorp Chrome extension. The extension automatically manages a user's Proxy Auto-Config (PAC) files that explicitly route special cases through the Access Proxy [2]. When a user connects to a network, the extension automatically downloads the most current PAC file and displays the good connection icon. Rules in the PAC file automatically route requests to corporate services through the Access Proxy. This allows our internal developers to deploy internal corporate Web services without explicitly configuring client access: they

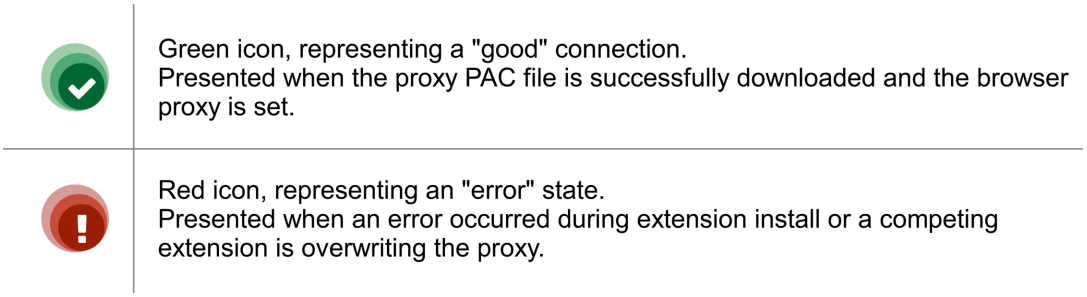

Figure 2: Icons in Chrome extension that indicate authentication state

deploy a service that will have a CNAME DNS entry in the public address space that resolves to the Access Proxy. The Access Proxy then automatically handles the user authentication and authorization.

Since the BeyondCorp extension routes all traffic through our Access Proxy, users can't communicate with devices that the Access Proxy can't reach. Additionally, the extension must be able to download a correct PAC file in order to route their traffic appropriately. This setup causes issues with common technologies like captive portals or when users need to communicate with devices on private local networks without routing through the Access Proxy. We needed a way to explain these scenarios and remediation steps to users, ideally without increasing load on Techstop. The Chrome extension's authentication state icons (shown in Figure 2) provide a gateway to further troubleshooting information.

When Things Go Wrong

What happens when things break or users run into complicated corner cases? By acknowledging that users will run into problems, we can identify the most common scenarios and develop plans to resolve them as smoothly as possible. Empowering our users to understand the problem and self-remediate when possible is our constant overarching goal.

Issues That Can Be Self-Remediated

Captive Portals

Because we're a global company with many traveling employees, users commonly encounter captive portals when working from airports, hotels, and coffee shops. These portals are usually implemented on the default gateway of a private network. When a user connects to this network, the BeyondCorp Chrome extension attempts to download the PAC file, but the captive portal prevents a successful download.

To resolve this issue, whenever the extension detects a network state change, we determine whether the device is behind a captive portal: we simply attempt to retrieve the Web page at http://clients3.google.com/generate_204, which is an empty page that always returns an HTTP 204. If we receive anything other than an HTTP 204 (most commonly, an HTTP 302), we assume that the device is connected to a captive portal. We then fall back to a predefined PAC file that we store within the extension itself and alert the user.

Users confronted with a captive portal can click on the Chrome extension icon, where we let them know that this issue is common when trying to authenticate to networks at airports or hotels. BeyondCorp is working as intended, and they just need to change the BeyondCorp setting to **Off: Direct**. Users can then complete the authentication through the captive portal, at which point the extension can successfully download the latest PAC file. This simple flow allows users to completely self-remediate with minimal downtime and no support load on our Techstop.

Local Network Devices

Users also frequently attempt to access devices on private address spaces. Many Google employees use their corporate laptops for tasks like configuring personal printers or other networking equipment. However, since we route all connections through the Access Proxy, access fails when the BeyondCorp extension is enabled. Similar to the captive portal use case, the solution is to change the BeyondCorp setting to **Off: Direct**. Unlike the previous case, we can't easily detect this failure state. Typically, users in this scenario have an active and functioning Internet connection. From the extension's point of view, everything is working normally and the user can access all corporate resources, so there is no reason to raise an alert.

To figure out how to effectively interface with users in this situation, we worked through a representative user journey: an engineer takes their corporate laptop home and wants to use it to change a setting on their home printer, which they connect to via its IP address. The user connects to their home network, and the BeyondCorp extension connects successfully, downloads the latest PAC file, and configures the browser's proxy. When the user enters the printer's IP address in a new browser tab, the request is sent to our Access Proxy along with all other private address space traffic. The routing request fails and the user gets an error.

We came up with a solution to this user journey by focusing on the end result: an error page from the Access Proxy. We created a custom HTTP 502 error message to insert into our error pages when certain conditions are met—specifically, whenever we return an HTTP 502 and the user was attempting to reach an RFC1918 or RFC6598 address. The error message explains to the user that if they were trying to access a local network device such as a home router or printer (the two most common cases we found), they need to switch the BeyondCorp extension to **Off: Direct**. In this way, we were able to use already existing infrastructure and processes to allow users to self-remediate the issue.

Custom Proxy Settings

Our employees sometimes need to set custom proxies to test ads in foreign countries. If a user installs multiple extensions that each try to set the proxy, the extensions collide with each other, which can confuse users and break their access to corporate resources.

We approached this use case with two solutions. First, we integrated foreign country proxy settings directly into the BeyondCorp extension. When users have a business need to egress from a specific location, they can select that location from a dropdown of supported countries directly within the extension. This provides our users a single extension that manages their most common business proxy needs.

Additionally, when a user has a valid need to run a secondary proxy management extension, their BeyondCorp icon switches from green to red. We then give them an option to change their state to **Off: System Alternative** and explain when they want to use this setting. Again, this process allows the user to self-remediate, increasing their productivity and reducing queries to our support teams.

Explaining Complicated Failures: The Portal

For simple cases, like those described above, we could empower users to self-remediate using quick customizations to our error pages or the Chrome extension. However, in cases of legitimate denials of access, we knew that users and support teams would want or need to know why they were denied. The complex, multi-layered ACL logic in our back-end infrastructure can make understanding the logic behind a specific decision difficult for users and support teams alike. It might take even a seasoned SRE multiple minutes of querying many internal services to identify the cause of a single 403 error page. Given the volume of 403 error pages served by our Access Proxy daily (~12M for HTTP/S alone), human involvement in troubleshooting is unscalable and impractical.

To facilitate diagnosing and troubleshooting more complicated BeyondCorp access issues, we designed a single portal to assist both users and support teams. Instead of just telling a user that they were denied access to a resource with a generic error code, we explain why they were denied and how to resolve the issue. The portal is standalone, rather than integrated directly in the Access Proxy, because it uses more granular ACLs that depend upon the end user's current trust level. Since the Access Proxy is available publicly by design, we wanted to limit the amount of knowledge an attacker can gain from the 403 error pages.

Architecture

The portal is roughly split into a front end and a back end, with an API that communicates between the two.

- The front end is an interactive Web service. It issues requests against the back-end API based upon input from the user.
- The back end can query multiple infrastructure services involved in access decisions. It deliberately omits various caching layers so users receive fresh information.
- The API between the front end and back end is also exposed for other uses, like batch processing and analysis, or embedding the output in other tools.

Explanation Engine

Beyond querying and surfacing ACLs, the portal also needs to present this information to users in a useful way. We built an explanation engine to provide troubleshooting details in response to parameters of deny requests. It operates by recursively traversing a tree of subsystems that provide authorization decisions.

For example, the Access Proxy ACL might require a device to be fully trusted in order to access a particular URL. Upon retrieving this ACL, the engine contacts our device inference pipeline to retrieve the conditions necessary to access the corporate resource. We then propagate this information to our front end and translate it into plain language, so the user can visit the portal to find out what's wrong with their current state and how to fix the problem.

ACLing the ACLs

While the explanation engine provides users with helpful information, the data it exposes can be sensitive. It reveals the problematic ACLs of protected systems and discloses information about the state of the user's account and device—all useful information for potential attackers. Defining the ACL for this data is a tricky process, as we need to balance tool usability against the need to protect sensitive information.

Depending on the user and device requesting troubleshooting information, we can replace sensitive nodes in the output with

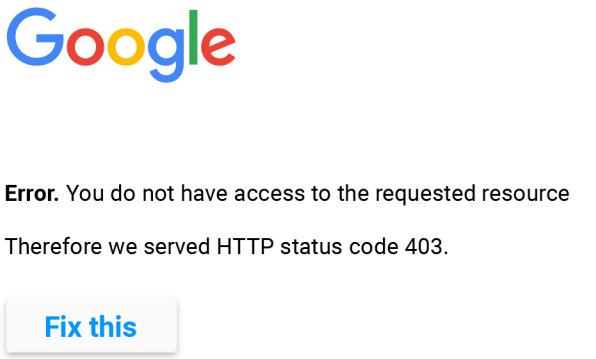

Figure 3: An error page displayed when BeyondCorp blocks a request

less specific variants. In extreme cases, we replace a node with instructions to contact our Techstop. In such cases, our Techstop and SREs can help users without disclosing sensitive information by verifying the user's identity and viewing the relevant information on their behalf.

Access Deny Landing Page

Once we developed the portal, we exposed it to users by integrating it into our Access Proxy error messages. When a user hits an HTTP 403 error, they see a button routing them to the portal, where we've automatically forwarded all relevant error details (see Figure 3). The portal then replays the access request against the back end and explains exactly what caused the issue.

For example, if a resource requires membership in a specific group, the portal provides the group name and a handy link to our group management system so the user can request access. Behind the scenes, the portal queries our back-end ACL services to determine the authorization requirements of the resource in question, and compares that information against the user's group memberships. The front end then converts the result of that comparison into a human-understandable statement (see Figure 4). This all happens in a matter of seconds, far faster than it would take the user to puzzle through group membership issues or reach out for assistance.

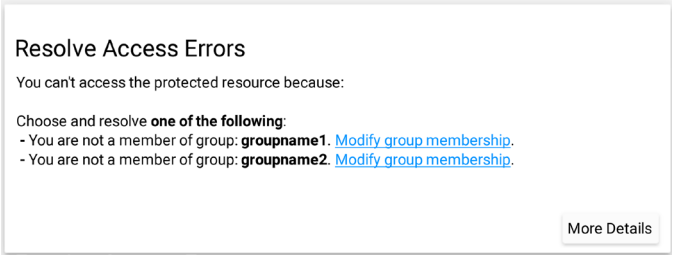

Figure 4: Employee-facing guidance on troubleshooting an access denied error

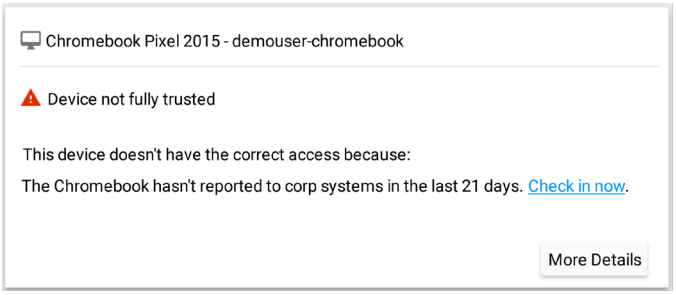

Figure 5: Service desk-facing guidance on troubleshooting an access denied error

Integrating this flow directly into our error messaging allows users to complete this process seamlessly and—most importantly—completely via self-service.

Ad Hoc Troubleshooting

Although we expect most users to access the portal through an error page, we also provide a direct page for more ad hoc troubleshooting. This landing page on our portal front end is customized according to the identity of the user and device accessing it. It presents information about the user and all their devices, and highlights issues that can potentially result in denial of access. By allowing end users to proactively visit this tool to get a global view of all of their devices and potential future access issues, we equip them to remedy issues with any of their devices in one fell swoop. This feature is particularly handy for checking device trust before a trip or demo.

Empowering Support

This front end also empowers our Techstop team to perform detailed troubleshooting quickly by providing immediately actionable steps, which dramatically reduce time to resolution. For example, to explain a 403 error page, techs can use the portal landing page to query for a specific username or device identifier. They can drill down into a specific device to determine whether it's a fully trusted corporate device. If it's not, we present the exact reasons why the device is not trusted and how the tech can resolve the issue (see Figure 5).

Future Goals

Beyond its current functionality, the portal also presents avenues for further automation. In the future, we plan to continuously run checks for potential denial of access issues. We'll notify users of any impending issues they can resolve on their own before those issues manifest in a detrimental way. Similarly, we'll identify critical issues that can't be self-remediated and automatically notify our Techstop with remediation steps. We also hope to expand the range of issues we can solve automatically without human intervention.

Focus on the Experience

Although the migration to BeyondCorp was challenging on multiple technical fronts, it allowed us the freedom to reevaluate our primary user support experience. By focusing on our users during and after the migration, we could deeply integrate processes and features that allow them to navigate the complex network model with ease. We designed our tools so that the user-facing components are clear and easy to use. These interfaces were purpose-built to allow self-remediation whenever possible, freeing up both user time and support channels. When users do need extra help, we provide tools and information to make our Techstop maximally productive.

For the vast majority of users, BeyondCorp is completely invisible. While Google employees worry about their own workflows, the model takes care of any and all access logistics. When users do have issues, we step in quickly and efficiently, giving them just the right information at just the right time to get them up and running again. Then we step back behind the scenes and let them focus on what they do best.

References

[1] B. Osborn, J. McWilliams, B. Beyer, and M. Saltonstall, "BeyondCorp: Design to Deployment at Google," *;login:*, vol. 41, no. 1 (Spring 2016), pp. 28–35: https://www.usenix.org/publications/login/spring2016/osborn.

[2] L. Cittadini, B. Spear, B. Beyer, and M. Saltonstall, "BeyondCorp Part III: The Access Proxy," *;login:*, vol. 41, no. 4 (Winter 2016), pp. 28–33: https://www.usenix.org/publications/login/winter2016/cittadini.

[3] J. Peck, B. Beyer, C. Beske, and M. Saltonstall, "Migrating to BeyondCorp: Maintaining Productivity While Improving Security," *;login:*, vol. 42, no. 2 (Summer 2017), pp. 49–55: https://www.usenix.org/publications/login/summer2017/peck.