



JASON MELLER

Honest Security

Endpoint Security & Device Management
That doesn't erode your values.



The Tenets of Honest Security

1

The values your organization stands behind **should be well-represented in your security program.**

2

A positive working relationship between the end-user and the security team is incredibly valuable and worth fostering.

3

This relationship is built on a foundation of trust that is demonstrated through **informed consent and transparency.**

4

The security team should **anticipate and expect that end-users use their company owned devices for personal activities** and design their detection capabilities with this in mind.

5

End-users are capable of making **rational and informed decisions about security risks** when educated and honestly motivated.

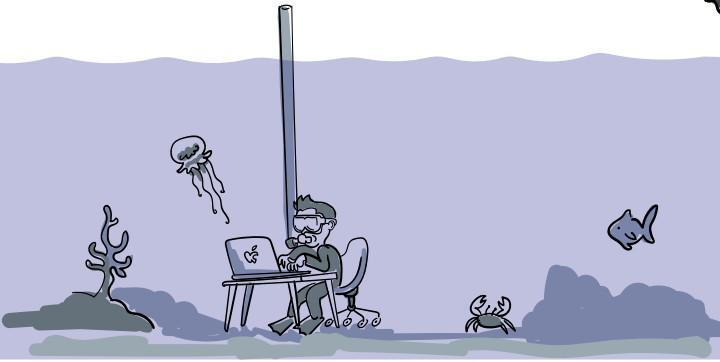
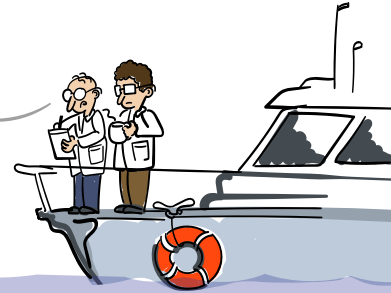
Table of Contents

1. Introduction	6
I Have Some Questions	6
The Tools Aren't Helping	8
Our Personal Lives are Inseparable from Our Devices	8
Why This Guide Exists	9
A Quick Note about the Examples in This Guide	10
2. "Why Honest"?	11
Dishonesty Stops You From Doing the Right Thing	11
You Aren't Dishonest. Your Tools Might Be	12
Open Source Doesn't Mean It's Honest	13
The Times They are a-Changing'	14
3. Setting Goals	16
Tricking Users Isn't Training Them	16
100% Compliance is Impossible Only Alone	16
Why Honest Security is a Better Fit	17
4. Collecting Data Honestly	18
The Big Question	18
Informed Consent	20
Transparency	22
The Insider Threat	24
The Importance of Ground Truth	24
5. Educating With Empathy	26
Recommendations Not Alerts	27
The Anatomy of a Well-Written Recommendation	27
Using Empathetic Intelligence to Divine Novel Insights	28
Recommendations Delivered at the Point of Performance	29

6. Achieving Compliance Objectives	31
Generating Predictable Consequences	31
Team Motivation, Not Gamification	34
7. Coaching the Culture Change	36
The Elites and the Underclass	36
Do Not Underestimate Fear of Change, Even Good Change	36
How to Coach	37
8. Author & Acknowledgements	38
About Jason	38
Acknowledgements	39

Honest Security

“Splendid— Inform the compliance team we’ve reduced the risk of fire.”



Introduction

I Have Some Questions

Are you a member of your organization’s security or IT team?

Good, keep reading, this guide was written for you.

Are you an employee that works for an organization that is using security and IT management software?

Good. Read this guide and find some members of your security team and share it with them.

In fact, before I start addressing the security team directly, I want to ask you, **the employee (or end-user)** some questions.

- When was the last time the security team gave you personalized advice?
- When was the last time the security team or even IT team asked for your permission before they made a change to your laptop?
- When was the last time the security team asked you for your thoughts on security risks you see in your daily workflow?
- Can you recall that time where they asked you how upcoming security changes to your device would impact your daily workflow?

Ok **security teams**, it is time for a reality check. As someone who has been in the security space for nearly a decade, I frequently chat with security and IT practitioners about their goals and objectives. Many teams describe themselves as an elite fighting squad working against shadowy forces who are trying to degrade, deny, and disrupt the key components of their organization. They adorn themselves with badges, flags, and other symbols associated with defenders and agents of authority. They firmly believe they are fighting the good fight, keeping the company's interests safe, and protecting their end-users from compromise or incident. They are the good guys.

It's always a surprise then that when I ask end-users the questions I posed earlier, the majority tell me the last time they talked with anyone from the security team was at a required annual training event. Some of them tell me they have *never* had a single one-on-one interaction with the security team. I've even met a few folks who have said their only interaction with the security team occurred when they clicked a link in a fake phishing simulation and were forced to take remedial training. They recalled how it made them feel humiliated. Others bemoaned the laggard performance of their once powerful laptops, rendered essentially unusable, because of all the numerous

security agents running in the background. That person confided in me that she rarely uses her work computer and instead writes all her work emails from her personal iPad.

These are not the hallmarks of a healthy relationship.

The Tools Aren't Helping

The products and applications which security teams use do not help improve this situation. Endpoint security is typically deployed under the cover of darkness where it remains mostly invisible to the end-user, persistently sapping the resources of the device for the benefit of the security team's detection mission.

These endpoint agents run with administrative privileges and give security teams complete access to the device. They can download documents, view the web browser's history, emit precise GPS coordinates of the device's location, install software, and even erase all the files. These functions occur silently in the background at the behest of an administrator issuing commands remotely.

Our Personal Lives Are Inseparable from Our Devices

Did your company buy you a Mac? If yes, do me a favor and open up the Photos app. Are you immediately greeted by your friends and family? What about the Messages app? Are there any non-work conversations found there? If you answered yes, you join the 90% of people surveyed by Accenture in 2018 who also claimed to use their devices for personal activities. 55% surveyed went on to say that they have allowed either a friend or family member to use their laptop. These devices are not simply tools for work, they are portals into our personal lives.

Despite the sensitivity of this data, the end-user is not kept in the loop about what occurs under the hood of their device. Security teams do not have to provide ongoing and informed consent for any of the surveillance capabilities these tools employ. While the security team *might* be accountable to a third-party group, that group *never* includes the end-user who is essentially being surveilled.

Why This Guide Exists

I created this guide because I believe that a healthy working relationship with the end-user is an essential asset, necessary for every effective security team. And because our industry is so obsessed with creating tools focused only on extending visibility and increasing the security team's control over the employee's digital assets, this critical relationship is irreparably being destroyed.

It's important for me to mention, not everyone has this wrong. In fact, there are dozens of security professionals whom I've spoken with that have healthy relationships with their end-users. The problem is that too few of them are talking about these loudly enough for others to hear. The approach of Honest Security is too important to be lost in the back-channel, so it is time to bring it forward where everyone can learn.

This guide is my attempt to codify the ideals and philosophies that we use as our north star at Kolide, a company dedicated to building products which foster this positive relationship. Some of the ideas presented here are a direct result of formative moments I experienced as an intelligence analyst at the GE CIRT. Other components are informed directly by my work developing many of the tools I deride in this guide in my role as Product Manager at Mandiant and Chief Security Strategist at FireEye. But much of the good stuff I didn't think up on my own. All of the advice worth listening to in this guide was hard-won based on the work the team and I have been doing since 2019 at Kolide when we decided to shift our focus onto the end-user.

My hope is that by publishing this guide, we can generate an important discussion about the state of our industry which ultimately results in the adoption of these ideals by practitioners, tool makers, and even commercial vendors. One company in this space is better than zero, but it is not nearly enough.

A Quick Note about the Examples in This Guide

Many of the topics in this guide can feel abstract without concrete examples. As mentioned earlier, much of this guide is derived from the experiences and insights I've had working on our product at Kolide. This guide is aspirational (and our product is still missing some pieces), but I will use examples from Kolide to demonstrate ideas throughout relevant sections.

While I hope people who read this see the value in our product, my intent is not to turn this guide into a simple sales pitch. Please afford me the latitude to use examples from Kolide today, but also reach out and share other products (or even better, open source projects) that capture the spirit of the examples. I will gladly add them to this guide in a future revision.

Why “Honest”?

We titled this guide Honest Security because we feel that the current approach practiced by most organizations can feel dishonest to many end-users.

Dishonesty Stops You From Doing the Right Thing

Do you work on a security team and think that’s an unfair assessment? Well, let me ask you a question; when was the last time you reached out to one of your colleagues to help them remove some adware or an evil browser extension that posed no threat to the company, but impacted their personal privacy?

To perform this simple service, a security team member must reach out to a user, admit there are tools installed monitoring the security of the device, educate them about this scope of this monitoring, and acknowledge that a human being is scrutinizing their installed apps and browser extensions, even when they affect personal information. After that’s done, you will need to finally ask for their permission to delete the app or remove the browser extension, a process that could perhaps impact their personal devices.

It’s no wonder this rarely happens. The friction involved in this helpful gesture is enormous. How do you help someone when in your first interaction you have to explain to them who you are, what your team does and expose the lack of transparency into the surveillance apparatus you use to perform your mission? The dishonesty in this case stems from lies of omission. We allow the end-users to believe whatever best suits them as long as it’s not disrupting the mission

of the security team. Filling that information vacuum with facts and information is a lot of work and it's not possible or even appropriate to begin that work the first time you have something important to say to an end-user.

This is the expected outcome of a dishonest approach. For many, this is how it's always been done, but that doesn't make it acceptable.

We call this new approach **Honest Security** because we fundamentally believe the benefits that the security team obtains from forming a working relationship with end-users can only be realized when that relationship is reinforced with accountability, transparency, and ethics. In other words, honesty. Without honesty at its core, the relationship has no future and therefore cannot have value.

You Aren't Dishonest, Your Tools Might Be

I want to clarify an important point, I think 99% of security practitioners in this space are honest. However, the tools that most security practitioners use, and the methods with which they use them can be categorized as dishonest. The reason we made Kolide was because we believe that if tools were available which helped improve the relationship between the security team and the end-user, security teams would see their value immediately. So far, it seems that bet was the right one.

With that said, if you are feeling defensive, it is important that you separate yourself and your personal ethics from the word "dishonest". Admitting the tools you have been using to do your job have dishonest characteristics does not make you irredeemably dishonest. On the other side, as an honest person, our hope is you see the value in the techniques described in this guide and will advocate for them in your organization.

As a passionate security enthusiast, I cannot truly express how freeing it was to implement these techniques and to shed the hundreds of pounds of cognitive dissonance I had built up over the years to protect my own ego. Nothing feels better than when everything is out in the open.

Open Source Doesn't Mean It's Honest

Transparency is a key part of honesty. This is the reason Kolide open-sources all of the code that runs on the end-user's device. It's also why our agent uses osquery, an open-source endpoint instrumentation framework created by the security team at Facebook, now managed by the Linux Foundation. Everyone has the right to know what code is running on their device, and we encourage them to contribute and improve that software for the benefit of everyone.

With that said, open-sourcing the code is not enough. Left unchecked, even a tool like osquery (which has contributing members who work diligently to find the right balance between user privacy, performance, and features) can be used for evil.

In osquery's case, these things include:

- Tracking someone's precise Geolocation (See table `wifi_survey`)
- Download/Reading the contents of any file on the computer (See table `carves`)
- Viewing conversation history of Slack, iMessage, and other sensitive chat tools. (See table `plist`)
- Locating files by their spotlight metadata (See table `mdfind`)

Abusing just one of these powers could be devastating, but chaining the above capabilities together with bad-intent can have despicable

results. As an example, with access to these tools on an end-user's device, one could instruct osquery to download all of the photos of the person they talk with the most on iMessage that were taken within a 300 ft radius of the location the owner of the laptop spends most of their time. Yikes. Worse yet, using an osquery front-end, an ill-intentioned security engineer could run that query across thousands of devices simultaneously, sifting through the results at their leisure.

The problem is, the developers of osquery, and osquery-based solutions, can easily rationalize why each of the features which facilitate the above scenario exist. They aren't doing anything wrong, but at some point there needs to be someone along the way thinking about how these tools can be used. Right now they assume that it's you, the security practitioner. Yet despite granting you that awesome responsibility, they offer little help in the form of tools or guides that will enable you to succeed. Ultimately, without built-in accountability to the end-users who stand to be harmed by osquery's misuse, these features that have independent merit can do some real harm. Honest Security practitioners can't fix these upstream problems, but they do have a responsibility to identify and mitigate them ¹.

As you can see, open-source does not inherently create things that can solely be used for good. If we are practicing Honest Security, it is our mission to create experiences that ensure the power afforded to us by these tools is used responsibly, and the people who wield them are held to account.

The Times They Are a-Changin'

This guide primarily centers its advice on the assumption that your company issues organization-owned devices to its users. We focus on this because it's a [steel man](#) argument against Honest Security. "Users should not expect privacy on a device they didn't purchase", is the common retort I hear from experienced IT and security professionals when they are faced with the ideas and recommendations

found in this guide. However, I want to point out that end-users are often able (or sometimes even expected) to use their personal devices to do their job. This can range from simply connecting your work phone to company email, to using their own laptop and connecting it to the company's MDM. This isn't a trend that is going away, in fact, it's more popular than ever. To some, using a personal devices to do your job is as natural as a carpenter bringing her favorite hammer to a work-site. When these personal devices are used, many organizations apply the same—or even more—rigorous management controls and security capabilities. I have even spoken to users who had their personal devices erased simply because it was part of an IT administrator's off-boarding process. If you fall into the camp thinking that Honest Security isn't something you want to try; consider how you might feel if trends continue the way they have for the last few years. We should be embracing a security strategy that will be compatible with these trends; not invest our time, budget, and team's resources in a strategy that is clearly in its autumn, not its spring.

-
1. While osquery is one of only a few endpoint agents that actually makes any consideration for the end-user's privacy, you might find this criticism unfair. However, I know many of the people on the core team. Not only can take the heat, but they are incredibly responsive to feedback. I know together we can get osquery and many other projects compatible with Honest Security.

Setting The Goals

Honest Security is an approach designed to help organizations achieve two distinct goals: educating employees about security, and dramatically improving adherence to the security team's recommendations and compliance objectives.

Tricking Users Isn't Training Them

In your organization, your team might employ tools and programs with the same goals in mind. Perhaps your organization even lists honesty, transparency, and ownership as its company values. On the education side, there are many products that allow you to create interactive training modules for employees. Despite their incredible graphics and engaging voice acting (not), these programs fail to deliver the information when it's most beneficial for the trainee to hear it, at the point of performance. Other educational tools like the ones used to train people to identify phishing emails *do* deliver the training at the point of performance (right after the user is fooled by a phish) but can only do this by expending considerable effort attempting to fool, entrap, and subsequently humiliate their potential students into demonstrating their lack of knowledge. That's not a great way to build up a working relationship.

100% Compliance Is Impossible With Automation Alone

On the compliance side, the existing approaches aren't much better. Products exist which allow you, with a single click, to remotely change all the settings of a device in order to match the recommendations for a given compliance standard. Unfortunately, these tools do not consider nor care about the end-user's experience, nor do they take

into account the context of their current environment. If a compliance standard advocates that bluetooth should be disabled in high-risk situations like when the user is in public spaces or traveling, then the compliance tool's only option is to disable the bluetooth for all devices, permanently, no exceptions. Need bluetooth for 10 minutes so you can join a conference call on your Mac with your AirPods? Too bad. That's how the hackers get in.

Why Honest Security Is a Better Fit

Honest Security helps achieve both of these goals better than the existing approaches because it believes that communicating directly with the individual who is using a device is the key to solving these problems.

As we will see in the Honest Security for Education section of this guide, we can use honest data collection techniques to deliver contextual and personalized recommendations (not alerts or failures) to end-users at the point of performance. No humiliation is required!

In the section entitled Honest Security for Compliance, we will explore how generating predictable and proportional consequences increases the adherence to important recommendations from the security team. The techniques described in this section can even drive actions like users voluntarily opting in to managed profiles.

But before we can do either of these things, our Honest Security program needs to be able to know some things about your organization, your users, and the devices they use, and we need to obtain this knowledge honestly.

Collecting Data Honestly

The Big Question

Let's start with a question I typically ask end-users.

Do you know **if** your organization looks at your web browser history?

I want to clarify that I am not asking them whether or not the company collects web browsing history. I am asking them whether they *know*, with 100% certainty, whether the security team is or is not. It is one thing to know whether your organization *can* view your browsing history, and another to know *if* they do. Also notice I didn't ask them if the organization *usually* looks. One person, looking once because they were curious, *is looking*.

After making these clarifications, it is my experience that there are three camps of people who can still emphatically answer "yes."

The Three Camps Who Say "Yes"

The first camp are the folks that definitely know they are being surveilled. They know because they have seen their peers reprimanded or even fired because of their web browsing history. In these organizations, the leadership is all too proud to communicate the veracity and insidiousness of the surveillance because they see it as a deterrent to people slacking off, stealing from the company, and other assumed nefarious activities their so-called "trusted" employees might be getting up to.

The second camp are the folks who know they are not being surveilled because the company has no security or IT tools or policies. On their first day they opened up their laptop, fresh out of the packaging, and did not have to install any company mandated security tools, management profiles, or agents. They work from home on their own Wi-Fi, and they never need to connect on a VPN. The great irony here is that these users might be ignorant there are capabilities like Apple's Device Enrollment Program (DEP) that are able to put Macs under management fresh out of the packaging directly from Apple's warehouse!

The third camp are the folks who can answer “yes”, because they know exactly what tools are installed on their devices and what the tools are capable of collecting. More importantly, they know they can *independently verify* how the security and IT team is using these tools in practice. They know if the security team is looking at their web browser history because the tools the security team uses *require* them to know. These are people who work for companies that practice Honest Security.

The Folks Who Say “No”

But what about the most common case, when end-users answer “no?” If you don't know if your organization looks at your web browser history, why don't you know? How can you find out? Who do you reach out to to find the answer? Do you feel empowered to ask these questions without reprisal?

If you pushed your organization to adopt Honest Security, this conversation will be easy to have. If it hasn't, then users are stuck in the dark, not knowing what tools and security practices are being used on devices. You may discount the psychic toll this has on end-users, but it's very real and it can negatively influence their behaviors and decisions, in the same ways excessive surveillance can when implemented at a nation-state scale.

Informed Consent

Informed consent is a term used by the medical community. In the United States (and likely abroad), if you've ever undergone even a minor procedure that carries some risk, a medical professional likely walked you through those risks, even if their probability is infinitesimally small. Not only did they do this verbally, they likely gave you a printed document that could not be adulterated by the medical staff.

Practicing informed consent in medicine is obviously the right thing to do, but it's not something that organically came to be. Its codification into Federal law is the result of hard won litigation by people permanently harmed by medical procedures where they felt they were not adequately informed about the risks. Before informed consent, those infinitesimally small risks were buried in legal documents with 8pt font. If discussed verbally, they were diminished by doctors pushing for specific outcomes that they thought were best. While withholding information may feel like you are just making choices clearer for people, it's essentially lying by omission, and this approach was dishonest enough that laws needed to be created to protect people from harm.

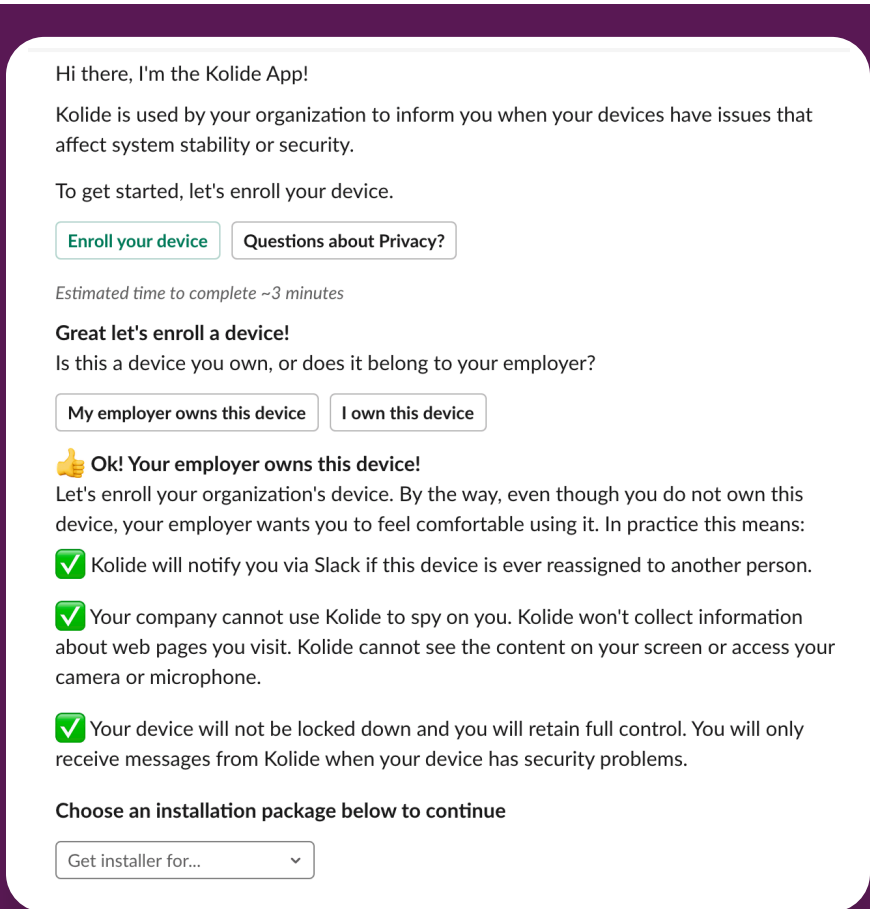
Switching back to modern endpoint security, the lack of informed consent continues to be a key feature. Security team members feel that the statement in their employment agreement, or the one bullet on slide 34 of last year's security training powerpoint are sufficient forms of consent.

In Honest Security, informed consent should take place in moments when there's a chance lack of consent could damage the trust between the end-users and the security team. An example of a situation where consent should always be obtained is the deployment of software that collects facts about user devices. As benign as these facts may be, users have the right to understand and control this process.

Example: Onboarding

Kolide's product obtains this consent process through an enrollment workflow in Slack (pictured below).

You'll notice a few things here. We aren't rolling out the endpoint security software silently in the background and informing users after



Hi there, I'm the Kolide App!

Kolide is used by your organization to inform you when your devices have issues that affect system stability or security.

To get started, let's enroll your device.

[Enroll your device](#) [Questions about Privacy?](#)

Estimated time to complete ~3 minutes

Great let's enroll a device!

Is this a device you own, or does it belong to your employer?

[My employer owns this device](#) [I own this device](#)

👍 Ok! Your employer owns this device!

Let's enroll your organization's device. By the way, even though you do not own this device, your employer wants you to feel comfortable using it. In practice this means:

- ✅ Kolide will notify you via Slack if this device is ever reassigned to another person.
- ✅ Your company cannot use Kolide to spy on you. Kolide won't collect information about web pages you visit. Kolide cannot see the content on your screen or access your camera or microphone.
- ✅ Your device will not be locked down and you will retain full control. You will only receive messages from Kolide when your device has security problems.

Choose an installation package below to continue

Get installer for... ▾

the fact. When embracing a consent based approach, the end-users are the ones who actually download and install the package. They feel in control because they are in control.

There are some additional situations that Honest Security requires consent for each instance. These include the following:

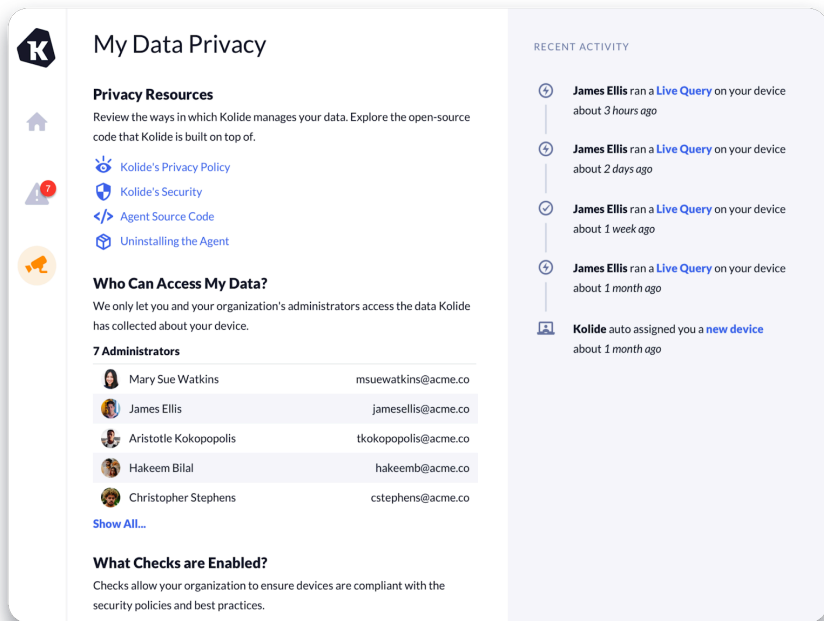
- viewing or transmitting the Geolocation of a device,
- reading the contents of files in the user's directory,
- and remote-locking/erasing personally owned devices.

While these situations are obvious choices to require consent, your organization may feel they aren't enough. The consent requirements should be fine-tuned with input from the privacy sensitive individuals of your organization, the above only represents the minimum bar that every organization should reach.

Transparency

It's not practical or healthy to ask for informed consent for every single action. Imagine having surgery and you having to consent to every move they are going to make, every medical device they are going to use, and review every possible contingency plan. This is a waste of everyone's time and creates permission fatigue that results in the blanket consent for any task.

Where informed consent is inappropriate, transparency is essential. Many security products feature audit logs, but those audit logs are rarely readily accessible to end-users who stand to lose the most when a bad-actor abuses their access to their work device. In order for the automatic accountability transparency brings to bear, it must be consistently applied to all mechanisms available to the administrators to obtain data.



Example: A Privacy Center

At Kolide, all users have access to a Privacy Center that allows them to scrutinize how the access they consented to earlier is being used by the various members of the security and IT teams.

In the absence of transparency, some of your more worried users are left to assume the worst, that the access is being abused or that they are being unfairly monitored. Can you blame them? In the post COVID-19 era, the news is littered with human interest stories about evil bosses abusing computer management and IT software to spy on their employees.

As many security teams know, the reality is much more boring. Why not show them the truth? It makes everyone (including the security team) feel at ease.

The Insider Threat

The most common argument I see against transparency is that it gives bad-actors within your organization an advantage. The rationale is that an insider threat might be able to identify gaps in the security team's detection capabilities and systematically abuse them to complete their mission. I disagree. As we all know security through obscurity rarely works. Also, it's much more likely that this transparency and regular contact will instead serve as a deterrent. Unlike end-users who are making unforced errors, malicious insiders are afraid of being caught. The more interactions they have with a team practicing Honest Security, the more uncomfortable they will get.

Part of Honest Security is trusting end-users because they are our colleagues. If you build a dystopian and cynical security program born out of fear, mistrust, and suspicion, then you will inevitably make your fellow-employees your enemies. The positive working relationship we are advocating for in this guide cannot exist under such a program. Only you can judge if that trade-off is ultimately worthwhile.

The Importance of Ground Truth

In order for Honest Security to function properly, it must have highly accurate facts about devices, organizations, and most importantly people. These ground truths allow Honest Security to *confidently* make *correct* security decisions/assertions and deliver them to the right people.

The “ground” in ground truth indicates it is information provided via direct observation by a trusted solution. This is necessary for several reasons:

1. **Accuracy** of the data is essential to the Honest Security mission. Honest Security practitioners must be able to immediately and directly correct any inaccuracies. It is unacceptable for the system

to draw incorrect conclusions from bad data sources. Relying on substandard data puts the credibility of Honest Security at risk.

2. As mentioned earlier, in order to be honest, ground truth must be collected with **informed consent**. Dishonest security solutions do not care if their data sets are ill-gotten as they never rationalize data collection with end-users.
3. Honest Security should be accessible for businesses at *any* stage to implement. Increasing the **total cost of ownership** for early stage businesses by forcing them to buy additional products to power Honest Security's core mission is unacceptable.

In the absence of these positive guiding principles, dishonest security looks to amass as much data as possible, in unfounded preparation that its value can be fully realized later.

Honest Security asserts that the list of ground truth to collect is *solely* driven by the needs of the education and compliance mission. This assures that Honest Security solutions can always fully explain to an end-user why specific data is needed and how it is used.

Educating With Empathy

The term “empathy” has unfortunately been diluted by years of marketing overuse, but it is a fundamentally crucial concept that cannot be ignored. The term captures the number one skill you need to employ as you build your Honest Security education materials.

 Jasons-MacBook-Pro • MacBook Pro (16-inch, 2019)

Failing Check: Unencrypted SSH Keys

Reason: Unencrypted SSH Key Detected

Why is this a Problem?

An unencrypted SSH key increases the risk that the key can be used by an unauthorized person to gain access to a privileged system. This is especially true if the key is inadvertently synced to your Dropbox, Google Drive, or a backup.

Since the minor inconvenience of encrypting an SSH key far outweighs the potential impact, we recommend encrypting all SSH keys stored on a system.

Required Action:

Encrypting SSH keys is a trivial process that should only take a few minutes. On Mac or Linux simply follow these steps...

1. Open Spotlight search via the following keyboard shortcut: 'Command + Spacebar'
2. Type [Terminal.app](#) to locate your Terminal application and hit Enter to launch.
3. Once the terminal is open, at the prompt type the command(s) listed at the end of this notification

You will be prompted to create a passphrase. We suggest you create a unique passphrase per key and store those passphrases in a secure/approved password manager like 1Password.

You may not see text being entered as you type your password in. Do not worry, this is normal security feature of the terminal and it is receiving your keystrokes.

 `ssh-keygen -p -f /Users/jmeller/.ssh/aws-prod-us-east-ec2222`

I've fixed it. [Check again](#)

[Contact Admin for help](#)

Education without empathy is education that is not personalized and becomes ill-suited for people who are from under-represented groups. Security incidents are often a “weakest link in the chain scenario” – they are unintentionally caused by the actions of uninformed employees. It’s imperative then, that security education is designed to reach everyone, no matter their role or technical background.

Recommendations Not Alerts

Since the primary actor achieving the organization's security goals is the end-user, Honest Security does not generate alerts, alarms, or failures, it only generates recommendations.

Recommendations embody the values of Honest Security because they do the following:

Do not come pre-loaded with negative value judgement (unlike the other terms)
Serve not only to instruct, but also inform and educate
Are an open invitation for further discussion and understanding
Expanding on the last point, Honest Security provides opportunities for productive conversations about recommendations. These opportunities should include contesting inaccurate or unhelpful recommendations, or even deferring recommendation to a more helpful time in the future. Nothing makes a recommendation feel more like a command than having no channels to appeal or discuss its applicability.

The Anatomy of a Well Written Recommendation

The following example is one of the more technical recommendations we use in Kolide and have it enabled for all of our own employees, not just the software engineers.

I want to point out a few key elements in this recommendation. First, notice how we explain the “why” using clear and concise terminology. We haven't dumbed down the language, but we also avoid using overly-technical terms that don't add any value.

Second, we include some language that short-circuits the most common arguments that a defensive user might position against the recommendation. “Oh *this* SSH key isn't really that important, I just made it as a test so it's fine if it's unencrypted.” Yes that might be true,

but we've made it so easy to encrypt. Why not just do the right thing and make it a habit?

Third, we include step by step instructions that are customized to their operating system and assume no prior experience. See the sentence, "You may not see text being entered as you type your password in?" This is an example of empathy in action. We are anticipating that this might be the first time the user has ever interacted with the terminal and guiding them through a situation where they might think something is going wrong.

Finally, we give them the precise command to enter into the terminal and a feedback loop to verify that they did it right. They get confidence that the problem is resolved and they even feel good that they were able to do it without any help.

But suppose they need help or they object to the nature of the recommendation, they are one button press away from contacting a human being and getting things sorted. These aren't commands to be followed, they are conversations.

Using Empathetic Intelligence to Divine Novel Insights

Empathetic intelligence is what practitioners of Honest Security use to divine security relevant insights about an organization, their digital assets, and the people who use them.

Existing cyber security intelligence is typically generated by building detection models derived by analyzing the patterns of attackers, their tools, and their modus operandi. This intelligence is essential, but it's not the whole picture. Empathetic intelligence is generated by carefully considering the common use-cases across different roles, identifying sources of risk, and working forwards to generate the detection model.

In order to do this well, Honest Security must leverage the empathy of a diverse set of human beings to generate these insights. By its very nature empathetic intelligence cannot be automatically divined by artificial intelligence or machine learning.

Using this very process at Kolide we've often created insights that unearth new problems other security tools are completely blind to. At one organization, they used Kolide to determine that their back-end engineers had a habit of reproducing production bugs locally by downloading a copy of the database. Most of them dutifully remembered to delete the file, but none of them remembered to empty their trash. Unless you actually sit down with a developer and watch them do this, you might miss that last detail.

Recommendations Delivered At the Point of Performance

As previously discussed in the Goals section, the primary issue with classic forms of education is that they deliver their recommendations out of context.

Most organizations have an IT acceptable usage document that they give to new employees. In my experience, many of these documents are crafted with care and contain valuable information that not only protects the company, but represents good advice for people to be following even in their own personal lives. While this type of all-at-once education style serves to check the box for various compliance standards, it's a terrible way to actually teach someone with any hope of retaining the information.

The best time to teach someone a corrective action is when they are actively making the mistake. Most folks would nap through a video presentation about how to tie a figure-8 knot for rock climbing when they know there is a snowball's chance in hell they would even be in

that situation. I guarantee you though that same person will be all ears when those instructions are delivered moments before they tie the knot that might save their life right before their first climb. That moment is the point of performance; the time that training has the maximum impact on the outcome.

Bringing this back to security, imagine you want your users to use a password manager. In fact, you don't want them to use any password manager, you want them to use 1Password because you've done your research and you think they have the best tool on the market and it matches your opinions on how to generate good passwords.

Adding this recommendation under the "Password Hygiene" section of the training powerpoint presentation three days *after* employees have already set their passwords is the right information delivered at the wrong time. Not only will it not impact the outcome, they won't even retain it.

However imagine a tool that reaches out to that very same user the moment they download and attempt to install a different password manager. This helpful recommendation explains that the password manager they chose was probably fine but they might be interested in the fact that their company pays for 1Password, a better product, and provides a helpful download link to get it right away.

This is relevant information delivered over a trusted communication channel that's already been established. It's coming from an automated system so it doesn't feel like you are actively being watched. In fact, it feels like a helpful service more than a tool that benefits the security team. Now the chances are much higher this user retains this knowledge even in future situations where the recommendation isn't triggered.

Timing is everything, and Honest Security is about getting this timing right. It uses the ground truth at its disposal to not only generate recommendations but to also deliver them right when they are needed.

Achieving Compliance Objectives

As stated earlier, Honest Security can be used to improve your employees' understanding of security through personalized recommendations delivered at the point of performance. While education alone can have a modest impact on improving the organization's adherence to their compliance goals, education alone is not enough.

Let's face it, even if you have perfect knowledge of security, that doesn't mean you are motivated, or even willing to apply that knowledge. This section discusses two techniques available to Honest Security practitioners to dramatically improve adherence to these recommendations and compliance objectives.

Generating Predictable Consequences

A few months ago I went for a walk with my wife, Amy, and my infant daughter, Lucy. After about 30 minutes strolling through the park, we went home. I walked up to my front door to find it slightly ajar. I had forgotten to lock it. I had a brief moment of fear that maybe I was going to swing the door open and all of our stuff would be gone. That fear gave way to relief a few seconds later when it was clear everything was just as I left it. Another month or so later, I forgot to lock the door *again*. Same situation, except this time I was less afraid anything went wrong, and I was right; everything was still ok.

Last week I needed Amy to meet me across town, but she couldn't find her copy of the house key to lock the door. Recalling the last two experiences, I said, "just leave it unlocked, it's fine. No one is going to break in," and she did. Just as before, I was right, and nothing bad happened.

Even though I am aware this is such fallacious thinking, I still fell victim to it. I know that if I did this enough times, the chance I would fall victim to a house robbery will eventually approach 100%. I relied on a sample size of only *two* personal experiences accidentally leaving my door unlocked to inform an *intentional choice* to leave it unlocked. This is the problem with risky behaviors: they catch up to you, and when they do, the consequences can be devastating.

This is not an educational problem. You could force me to take training modules on the statistics of house robberies for houses with unlocked doors and it wouldn't have changed anything. The problem here is in the executive parts of our mind. The part that applies that knowledge into action even when there are competing priorities.

Consistency Is The Key

This lack of consistency in realizing negative consequences is endemic in the security space. It's this very same reason why, over time, our vigilance will slip. Keeping up with the security team's recommendations takes time, and when time is short, are you really going to spend it making sure you've applied the latest macOS updates? You've forgotten to stay on top of that dozens of times with no obvious problem, so it's not really an issue.

This isn't a new insight. In fact most IT administrators cynically assume this will happen. This is why many of them seek out tools that short-cut around end-users and just take care of this stuff for them. You can't forget to install your updates if they are automatically installed by the IT team. Even if it was possible to automate everything with management software (it's not), this approach generates serious usability problems. Suddenly the process that restarts your computer for that update kicks off in the middle of a sales call. Another time, you realize you can't turn off your firewall for a few seconds to see if that is what is causing Zoom not to connect. Then the last straw, as you read this guide alone in your house, your screen turns off and the

computer locks itself because you forgot to move the mouse for 90 seconds.

Honest Security takes a different approach. Instead of immediately looking for ways to extract human beings out of the compliance problem, Honest Security looks to generate consistency through proportional and, most importantly, *predictable* consequences when security recommendations are not followed by the end-user.

Steps To Create Fair Consequences

To reiterate, the steps below outline the most effective way for this process to work:

1. Articulate the nature of the consequence when delivering the recommendation
2. Ensure the consequence's impact on the user is proportional and relevant to the risk they are generating by ignoring the recommendation. (For example, it would be inappropriate to lock the user's email because they forgot to turn on their firewall.)
3. Always have an automated system follow through with activating the consequence immediately once the time limit has expired.
4. Ensure the user knows that the consequence has been activated and is given a clear and automated path to resolve the problem.
5. In the case where the consequence is particularly disruptive, (e.g. losing access to the VPN for failing to install updates) make sure you give the end-user access to members of the IT and Security team to temporarily lift these restrictions.

Opt-in Management

While this process is effective, there are just some people who will continually find themselves always on the brink of the consequence activating (or worse, serial offenders). In some situations, these users

may do much better with the recommendations they regularly fail to implement on time if the security team could just do it for them. This is where Honest Security can allow the users to opt-in to traditional device management solutions (where applicable) and not have to worry about getting locked out of critical services or accounts.

Team Motivation, Not Gamification

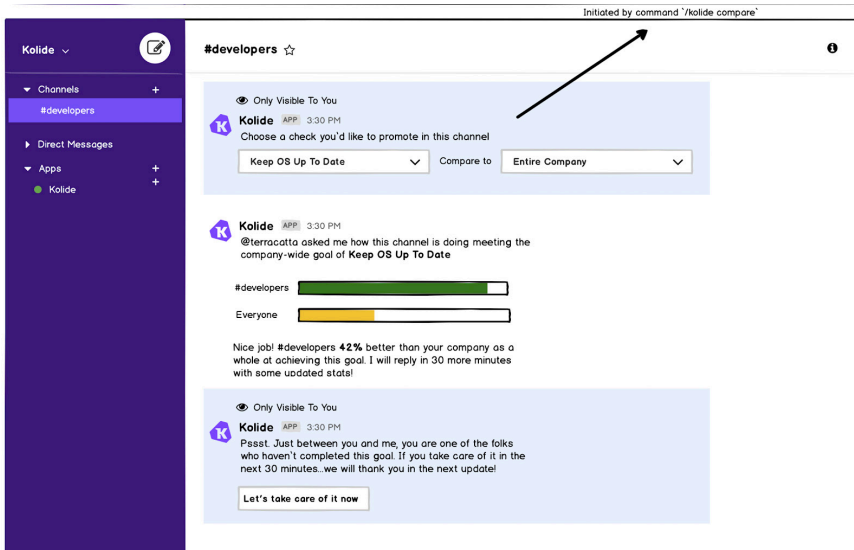
While generating consequences is the most effective way to quantitatively improve compliance, Honest Security also applies other techniques to help bring a little more respect and a sense of camaraderie to the experience. This is important because it adds a group dynamic to the relationship between the security team and the end-users, instead of solely relying on mostly automated one-on-one interactions.

Don't create individual winners and losers

Team motivation is an incentivization structure designed to create positive and socially visible rewards for groups adhering to Honest Security recommendations. Unlike gamification which compares individuals' achievements to ultimately create a dichotomy of individual winners and losers, Honest Security creates incentive structures at a team or group level. This allows competitive individuals within a group to rack and stack the group's achievements against the organization. During the camaraderie of racking and stacking, Honest Security engages and nudges individuals privately to improve their group's performance towards a particular goal.

Example: Slack Implementation

The next image demonstrates some early UX R&D in how a platform like Kolide might bring this concept to fruition in a Slack interface.



Notice how in this example, anyone can initiate a comparison between different teams/groups, not just a member of the security team. Also notice that the people who are detracting from the overall score are not shamed publicly. They are nudged quietly in private and given an opportunity to work through their recommendations with the incentive of being publicly thanked 30 minutes later when the statistics are updated.

This is a powerful way to encourage competitive members of your organization to drive action towards completing your security compliance objectives, while making security a visible part of the organization's culture.

Coaching

The Culture Change

The Elites and the Underclass

When some of our early customers deployed Kolide for the first time, something happened that we didn't expect: a few of the end-users were disgruntled. We realized that some organizations' existing approach actually created a dichotomy between people who had fully monitored and managed devices, and a few groups of elite individuals that were exempt from this process.

In organizations where people could choose their OS we saw a trend where end-users who chose Linux did so not just because they loved using it, but also because it allowed them to avoid the IT team installing their standard toolset (much of it being incompatible). Kolide however works great on Linux and now, for the first time, these people were asked to lose their special status they worked hard to obtain.

Do Not Underestimate Fear of Change, Even Good Change

Deploying Honest Security is a process designed to create positive change in your organization, but like any big change, it must be managed and tailored to your organization's specific needs. If you only rely on the automated parts of Honest Security you are not providing an avenue for end-users to express their concerns and for them to feel heard. This is where coaching comes into play.

Coaches are people with important responsibilities. They set the tone

for the Honest Security program and serve as the point of contact to hear and respond to end-users with concerns.

How To Coach

In our experience, a successful coach is capable of transforming their organization into a place that practices Honest Security. A coach will be successful if they do the following:

- Treat employee productivity and happiness as first-class objectives along with the security team's goals.
- Create opportunities for end-users to express feedback about the Honest Security program and respond to that feedback with empathy and understanding.
- Recruit knowledge experts in the organization to help define empathetic intelligence use-cases to further increase the value of Honest Security.
- Lead by example by ensuring that their security recommendations are always resolved in a timely manner.
- Recognize that certain people benefit from the existing dishonest structure (individuals who have been given special exceptions to the existing security tools) and work with them to make sure they are included as equals in the Honest Security program.
- Create a sunset plan for security software that does not adhere to the principles of Honest Security.
- Publicly celebrate significant achievements early in the Honest Security program roll-out.

The techniques in this document are not meant to be a rigid recipe, and having the right people to carefully adjust the program to the needs of their organization is essential for it to be successful.

Author & Acknowledgements

About Jason

This guide was written (with a lot of help) by Jason Meller.



Jason is the Co-Founder and CEO of Kolide, a startup that helps organizations implement the principles found in this guide through a beautifully designed Slack app.

Jason began his professional security career in 2010 when he started as a Cyber Threat Analyst defending General Electric's networks from persistent nation-state sponsored actors.

After GE, Jason has worked at Mandiant, Threat Stack, FireEye, and now Kolide where he has dedicated career to building and shipping products that help security and IT practitioners.

Jason codes in Ruby, and builds all his web apps and products in Rails.

Jason lives in Cambridge, Massachusetts with his wife Amy, his daughter Lucy, and his tuxedo cat, Belly.

- [Contact Jason](#)
- [Follow Jason on Twitter](#)

Acknowledgements

While I believe that the precise configuration of ideas and philosophies presented in this guide are novel to the majority of readers, they certainly are not original. Honest Security would not be possible without people generously sharing their techniques and ideas. I specifically want to call out the following people who have contributed significantly to the ideas embodied in the Honest Security guide.

Jesse Kriss - Netflix

Jesse Kriss' work promoting [Netflix's User Focused Security \(UFS\)](#) approach was the single biggest influence on my work at Kolide. Our discussions over the years about user respecting security tools have informed this document. Jesse and the rest of the UFS team had the vision to open source the tools they use to implement UFS internally at Netflix.

Jesse and the team continue innovating in this space. Netflix now has a dedicated User Focused Security Engineering team, led by Nicole Grinstead. Jesse and Christina Camilleri also recently gave a talk at QCon on [User Adaptive Security](#) which discusses the adaptations of their existing methodology to some of the new constraints thrust upon us by COVID-19.

Jeremy Daer - Basecamp

I also want to thank Jeremy Daer who works on the Security, Infrastructure, Performance team at Basecamp. Jeremy created a tool at Basecamp called [Shipshape](#) that embodies many of the values presented in the guide. In late 2018, when I was looking for people who would share their experiences deploying security strategies that aligned with their company's value system, Jeremy spent hours on the phone walking me through Basecamp's approach. I cannot thank him enough for that time.

Geoff Belknap - LinkedIn

While Geoff was the CISO of Slack, his work leveraging [Slack to do distributed security alerting](#) heavily inspired Kolide's focus on chat-based interactions to help foster the relationship between end-users and the security team.

Other Acknowledgements

A big thank you to Wailin Wong, one of the hosts of the [Re:Work podcast](#) for allowing me to discuss our approach at Kolide and the merits of user focused security.

A massive thank you to Ron Eddings and Chris Cochran from the [Hacker Valley Studio podcast](#) for allowing me to talk about and promote this guide on their show.

Last but not least, thank you to the hundreds of Kolide customers for embracing our vision of Honest Security and providing so much of the feedback that elevated this guide beyond an academic musing.