

Zero Trust is not Zero Effort: Planning your Zero Trust strategy

Graham Gilbert - MacSysAdmin 2023

> **whoami**

I'm Graham

Client Engineering at Airbnb

londonappleadmins.org.uk

macadmins.io

graham.at/movember

Hi! I'm Graham, and I'm a Senior Tech Lead Manager at Airbnb on the Client Engineering team. It's been absolutely wonderful to see you all in real life, so thank you to Patrik and all of the rest of the macsysadmin team for bringing us all together again after so long apart.

I am one of the co-founders of London apple admins, I am on the board for Mac admins open source, and I am also a testicular cancer survivor and a Movember fundraiser.

Testicular cancer is the most common form of cancer in young men - so common in fact that I've been contacted by two more people this year in our community who are going through their own battle with the disease. The Mac admin community isn't very big, so that goes to show how common it is. Cutting off someone's testicle isn't really a cure. It's pretty medieval if you ask me. The only way we are going to find a cure for this is by funding research and awareness campaigns. So gents, please check yourself, and if any of you find this talk useful in any way, please consider donating. And today we are going to talk about planning your Zero Trust strategy

**US General
Services
Administration:
Zero Trust**

- Authenticate, monitor, and validate user identities and trustworthiness.
- Identify, monitor, and manage devices and other endpoints on a network.
- Control and manage access to data flows within networks.
- Secure and accredit applications within a technology stack.
- Automate security monitoring and connect tools across information systems.
- Analyze user behavior and other data to observe real-time events and proactively orient network defenses.
- Support IPv4 and IPv6.

The US government defines zero trust as this {SPEED RUN}:

- Authenticate, monitor, and validate user identities and trustworthiness.
- Identify, monitor, and manage devices and other endpoints on a network.
- Control and manage access to data flows within networks.
- Secure and accredit applications within a technology stack.
- Automate security monitoring and connect tools across information systems.
- Analyze user behavior and other data to observe real-time events and proactively orient network defenses.
- Support IPv4 and IPv6. {BIG BREATH}

Look at that wall of text. It's huge. And I'm going to be honest here, it's pretty wishy washy language, and if you are basing your entire zero trust strategy on these bullets, you probably have no real idea of where to start.

Some incorrect ways to plan your Zero Trust strategy

So we're here - someone (maybe even you) has decided that you need to do "zero trust" **{emphasize the air quotes}**.

Before I get into how I think these things should be created, I'd like to take a quick look at a couple of ways to plan your zero trust strategy that I think will likely end in disaster.

Zero Trust in a box

Buying a Zero Trust product

If you have a less than mature security team, they might well just buy a zero trust product and leave it at that. You know, one of the VPN replacement things that promises access only by “managed devices” (which usually just means has been enrolled in your mdm at some point) or some sort of conditional access product that you can hook into your identity provider. This might get you closer to Zero Trust, but this should form a small part of your zero trust strategy. If it were so simple that we could just buy a product and call it a day, I’d have a much easier job and you wouldn’t be sitting here listening to me talk about this.



If you have a better security team, they will spend months working on the strategy in secret, and eventually present a bible sized document outlining every single detail of your organization's zero trust strategy and then they will tell you to do the bits they need you for.

Okay smarty pants, how
should we do this?

I've spent a little while talking about ways I've seen this done poorly before, I guess we should crack on with how I think this should be done.

**“You’re not a security engineer or a
project / product manager, why are you
qualified to give this talk?”**

You, the audience

Before we carry on, lets tackle the elephant in the room. I imagine more than zero of you will be sitting there wondering why this person who leads a client engineering team is standing here talking about how to plan your zero trust strategy



Zero Trust is a journey that involves your whole organization

A good Zero trust strategy needs input from many teams - if it is solely being run by your security team, then you will **not** get an effective strategy

So your first step is to think about your major stakeholders. You will need representation from your security teams, and likely your client engineering or CPE team and your identity team - enough people to get a good idea of your future needs, but not so many that you can't make decisions - you can always bring more people in later on if you need to.

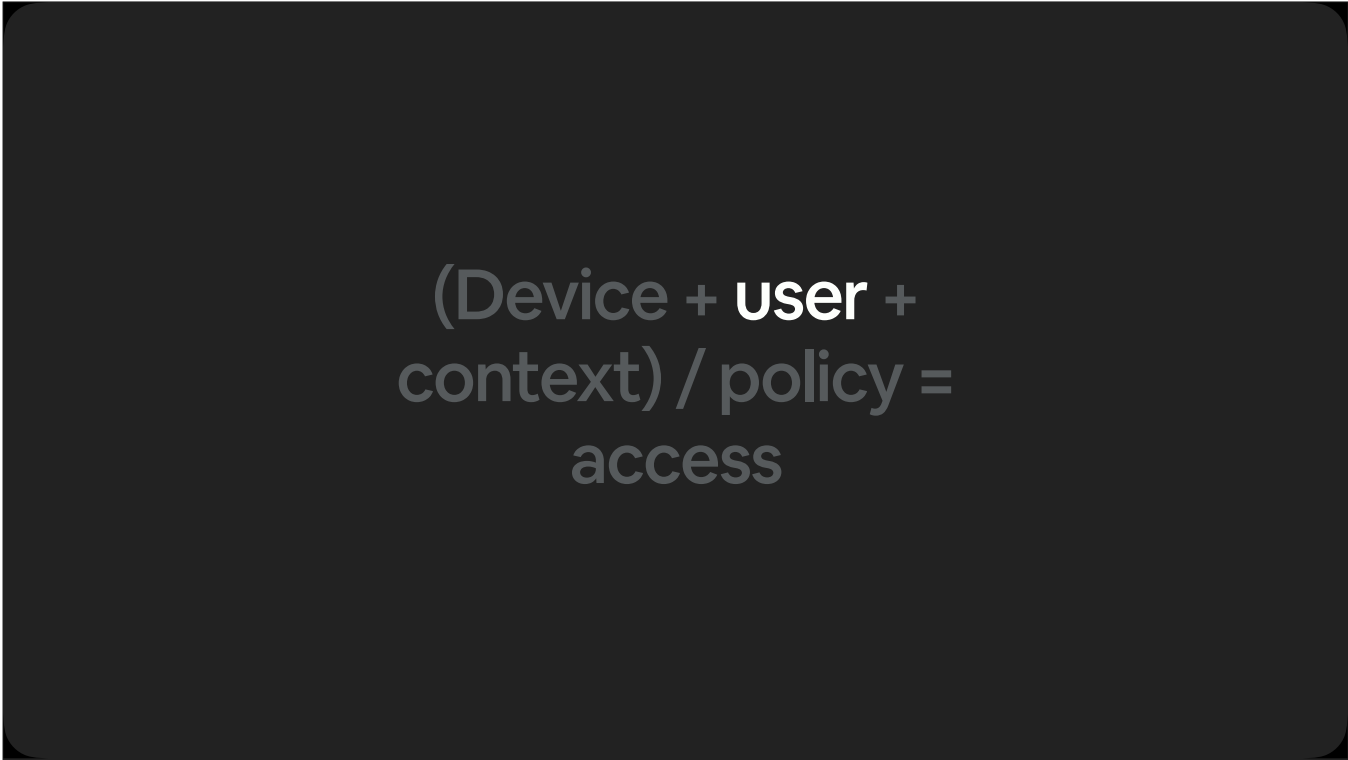
I have been one of those stakeholders. I have been the person who said to the security team "wait - I think we need more people to be involved here". Don't be afraid to do that.

$$\text{(Device + user + context) / policy = access}$$

I've gone over the official definition of what Zero Trust is - however, I think it's a very long winded and doesn't really say much, for what is essentially a simple concept. I'd love to take credit for this equation, but my manager came up with it. We were chatting away in our 1:1, trying to come up with the simplest way to describe what zero trust is to people who don't really know what we're trying to do and he pulled up a pad of paper with this written on it. It was like lightbulbs came on - this was it

$$(\text{Device} + \text{user} + \text{context}) / \text{policy} = \text{access}$$

The first part is the device. I don't mean just "approved, managed" devices - it is any device with some level of trust applied to it. The device might be a fully managed organization-owned laptop, fully patched with the latest operating system, with all of the security tooling working perfectly, or it could be a 10 year old android that got it's last security update in 2015 that is riddled with malware. It is everything we know about the device, good and bad.



(Device + **user** +
context) / policy =
access

Next up we are looking at the user - who the user is, what groups they're in, did they authenticate using modern MFA like a security key or a passkey? Are they using password-less? Did they use biometrics? All those sorts of things.


$$(\text{Device} + \text{user} + \text{context}) / \text{policy} = \text{access}$$

The context the request is in is what separates an okay zero trust solution from the great ones in my opinion. What contexts you care about is up to you - perhaps you do not want users making requests from certain countries, so the context we care about here is the location of the request. Perhaps the data classification matters to you - you could have different policies for PII vs publicly available data. Context changes constantly, and a good strategy will not only account for those changes, but embrace them to make good security decisions.

$$(\text{Device} + \text{user} + \text{context}) / \text{policy} = \text{access}$$

Finally we have the policy. We will take the sum of the device state, the user identity and the context that the request was made in, and then compare it to our policy. Your policy will likely be something along the lines of “the user must be in one of these groups, using a device with this level of trust, and meet these contextual requirements”



(Device + user +
context) / policy =
access

If the combination of device, user and context satisfies our policy for that particular resource, access is granted.

User Stories

So talking through that, it is beginning to sound like we are talking about user stories

User stories

As a <type of person>, I want to
<perform action>, so that
<benefit or outcome>

So here we have a user story. “As a **some sort of person** I want to **do this thing**, so that I **get some benefit or desired outcome**”

A user story allows us to talk about features or actions that people need to take, and justify why we think we should care about them. The purpose here is to put ourselves into the shoes of our most important customers - our users. If your users aren't happy with the solution you provide, they'll probably work around your tools and shadow IT it

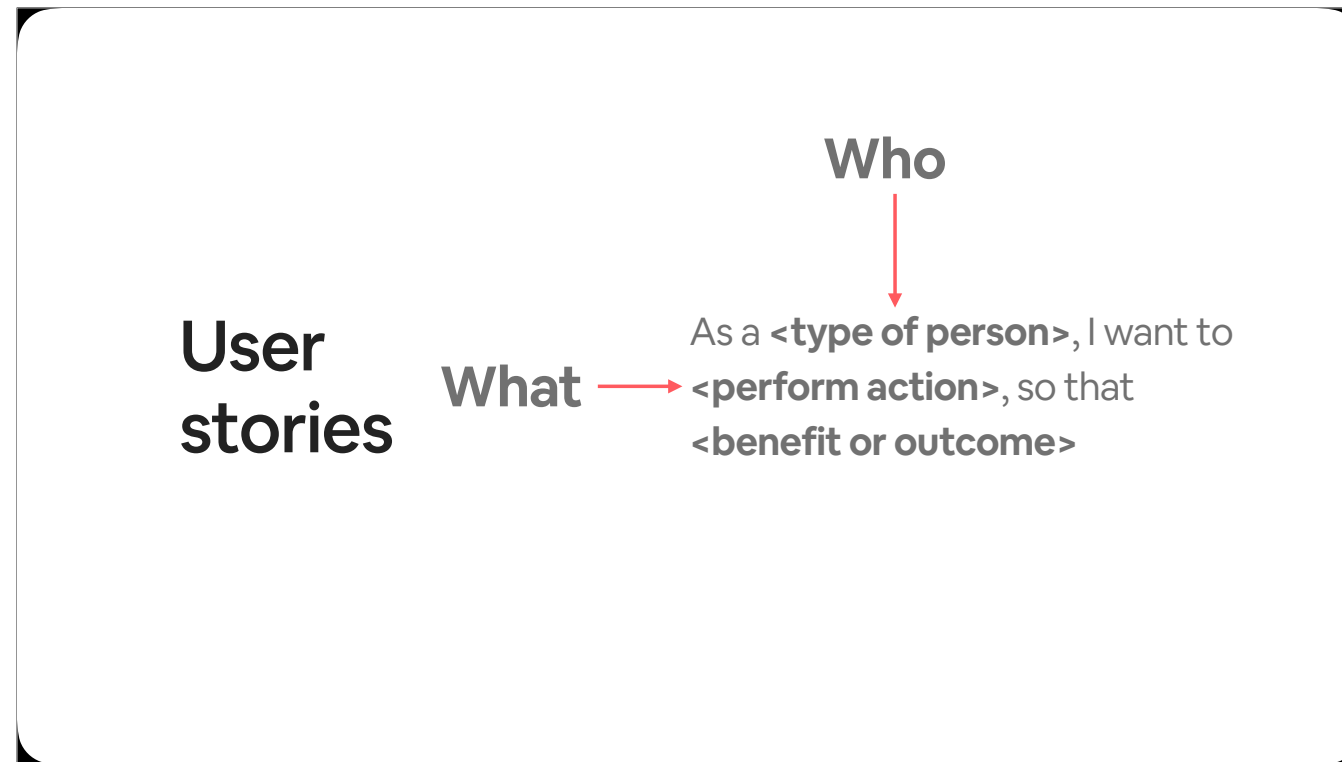
User stories

Who

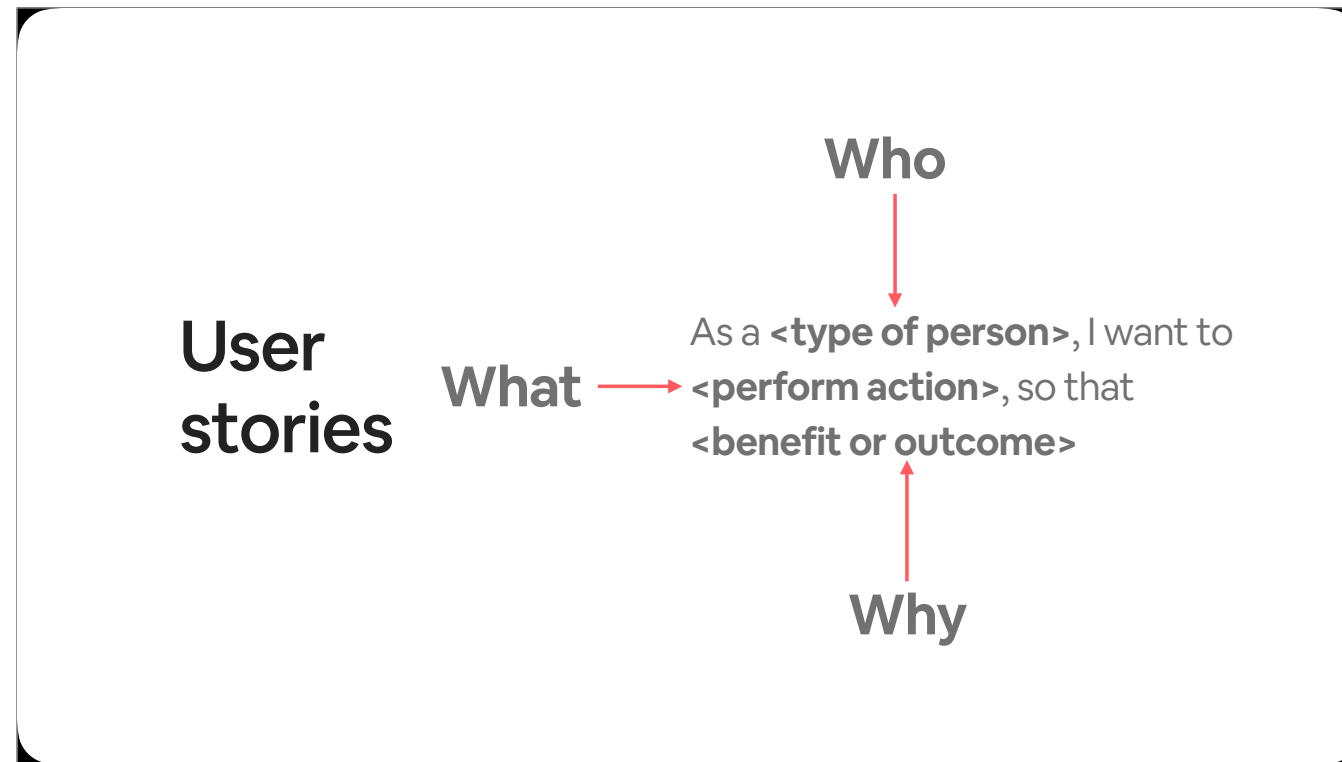


As a **<type of person>**, I want to
<perform action>, so that
<benefit or outcome>

So first off we have the Who - you should be reasonably specific where possible. Just saying “as a human being” doesn’t really tell you who cares about the feature you’re describing, so you could say “as a user who works in the accounting department” or “as a contractor”.



Next up is the what - what do they need to do? It might be access resources off the corporate network easily, or it might be know what version of macOS is acceptable in your patch policy.



The why is In my opinion, the most important part of a user story. It is why we think users want to perform the action, and as such it can act as a rough prioritization guide when planning on what to work on.

User personas

When talking about the “who” in user stories, you may find it helpful to focus on a few specific user personas. Your organization might already have some defined, so you should definitely re-use them if they exist. If you don’t, think of three or four types of people - outline what they do for their job, what tools they use, where they are physically located. Ideally you will have a broad cross section of your user base.

For our personas, we outline what they do for their job, what tech stack do they use, where they are, what internal and third party tools they use, and what their abilities are - are they technical and able to work around small issues, are they skilled in their field but not in a traditionally tech-y way etc

User personas

Ian: Back-end Engineer (FTE)

Priya: Designer (CW)

Maria: Customer Support (CS) agent

So these are the three people we use as a persona internally

We have a full time employee backend engineer who uses android

We have a contingent worker designer who uses iOS

We have a customer support agent - they will be on Chrome OS

User personas

Ian: Back-end Engineer (FTE)

Priya: Designer (CW)

Maria: Customer Support (CS) agent

Robot: Service to service communication

For our purposes, we also wanted to capture zero trust user stories for service to service communication - not user facing, but definitely a use case we needed to capture if we are not trusting anything on a network - we are still going to have services running in various clouds and SaaS etc that need to talk to each other

“As an FTE engineer, I want to be able to commit and deploy code without being on VPN so that I do not need to wait every morning for it to connect.”

Ian: Back-end Engineer (FTE)

So we've looked at what user stories are, and what our personas will be - let's look at some real world examples that we came up with. The first is “as an FTE engineer, I want to be able to commit and deploy code without being on VPN so I don't need to wait for it to connect”. VPNs are a real source of user frustration and friction. This isn't something that can be worked around, they either get onto VPN every morning or they don't ship code.

“As a CW designer, I need access to the files related to the project I am working on automatically on my first day of work so that I do not waste my first few days getting access.”

Priya: Designer (CW)

Here is another example of a user story - contingent workers face some of the same challenges as full time employees, but they have others too, such as that they are usually onboarded and off-boarded in a much shorter time period than full time employees. Getting them access to the resources they need to do their job on their first day of work without someone going in there and manually sharing them is a real issue.

When you are writing your user stories, you may find that you are writing down things that already work great and cannot really be improved - that's good too, you should write them down. We need to make sure that any of the controls we implement later on down the line don't break any workflows for our users.

Known tools

You probably are doing a lot of zero trust already, or already have tools that if configured correctly could be used as part of your strategy.

Write them all down, what they do, who owns them - try to categorize them too in clumps that makes sense to you - things like device management and identity

This will likely include your mdm, your identity provider, your MFA tool if that's separate, your VPN, your security software, things that issue certificates, asset inventory - those kind of things. Anything that is related to the device, the user, how the user accesses resources and how you assess the state of the device security

What are you trying to achieve?

From a security perspective, what are you trying to achieve here?

What are you trying to achieve?

- Block access to our IDP from unmanaged devices
- Provide user facing context reporting
- Continuously authorize access to services and data

Here are some ideas - you will hopefully have more goals than these, but as a kick off, you might want to

- Block access to idp from untrusted devices - ensure only devices you have decided are in a good state can access your services
- User facing context reporting to allow users to self-remediate issues - when you start making access decisions based on device and user context, your users are much more likely to be blocked. You may want to build or buy a tool that allows users to know what is wrong with their context and how they could remediate it
- Continuously authorize access to services and data - One of the core features of zero trust is to continuously asses the user, device and context against your policy, not only at authentication time but regularly throughout the session, and ideally with every request they make.

Pie in the sky ideas

Chances are you've got some really smart people together to talk about this - you should spend some time talking about ideas that you don't plan on tackling in your initial rollout, nor may they even be possible with today's technology. You should definitely write them down because they may form the basis of future projects.

Some examples we came up with

- Use AI to automatically share documents with a user when they're starting a project, and un-share them when they've finished
- AI to analyze user behavior - are they doing anything "weird"?
- Self service identity verification for password and MFA resets

Yes, lots of AI - when we had our people in the room, chat gpt was the new hotness

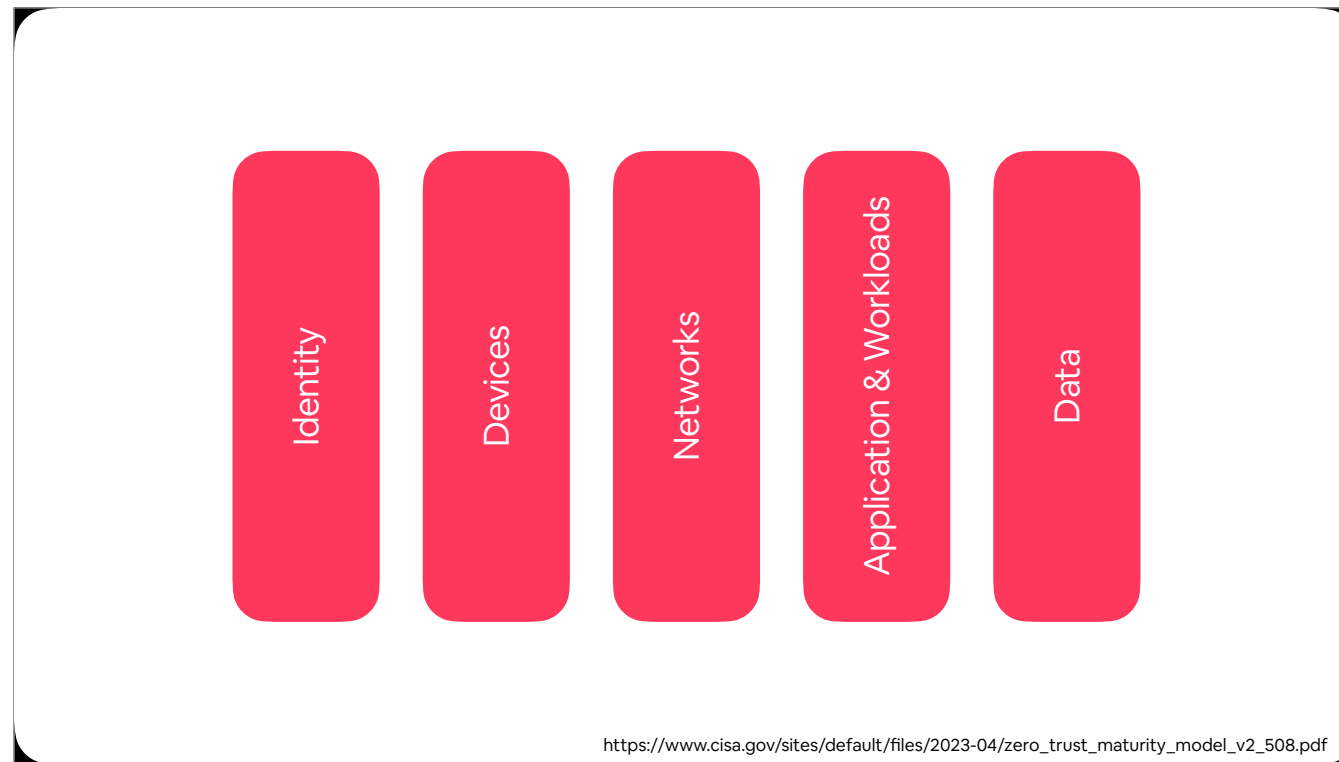
How will you measure success?

Measuring success is important for any project - how will you know when you're done? You could spend your entire life endlessly tweaking things to make them more zero trusty

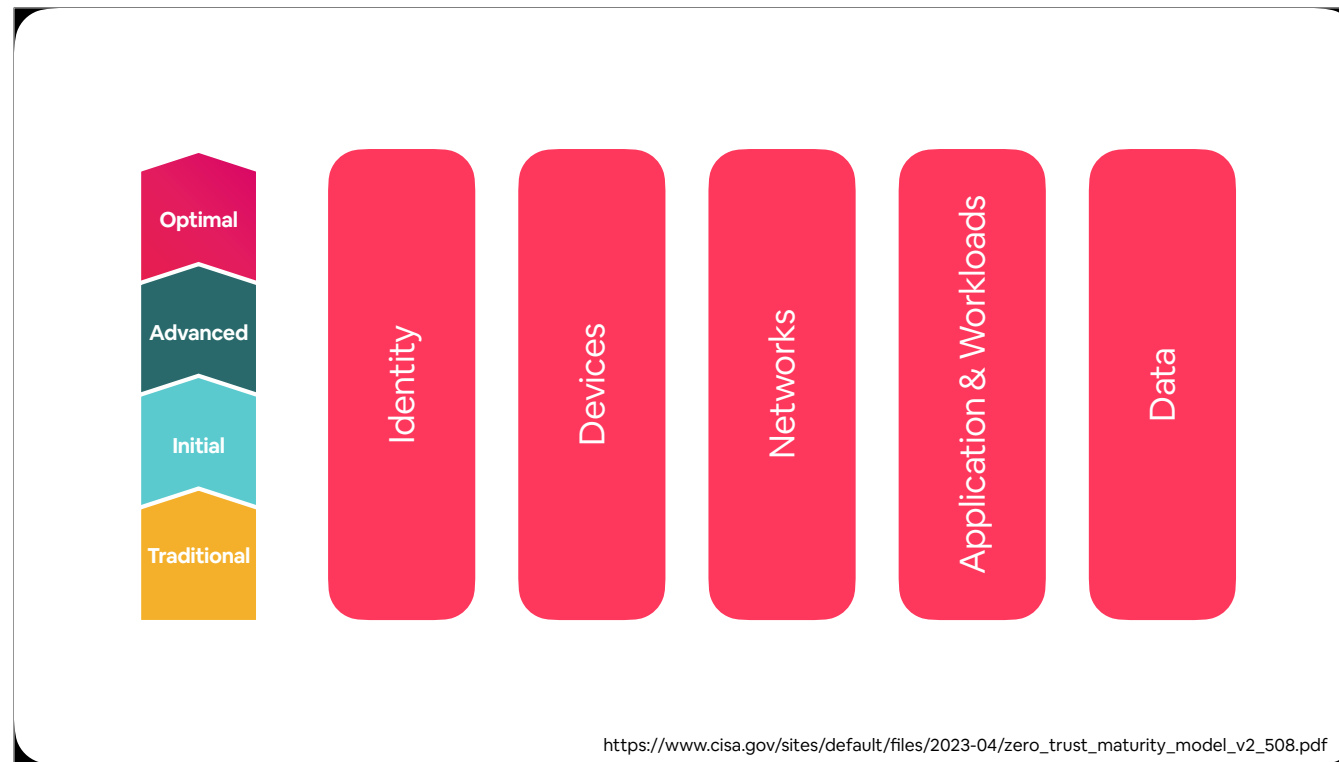
Maturity Model

https://www.cisa.gov/sites/default/files/2023-04/zero_trust_maturity_model_v2_508.pdf

Fortunately, CISA (the Cybersecurity and Infrastructure Security Agency) has published their Zero Trust Maturity model - what is a maturity model I hear you cry?

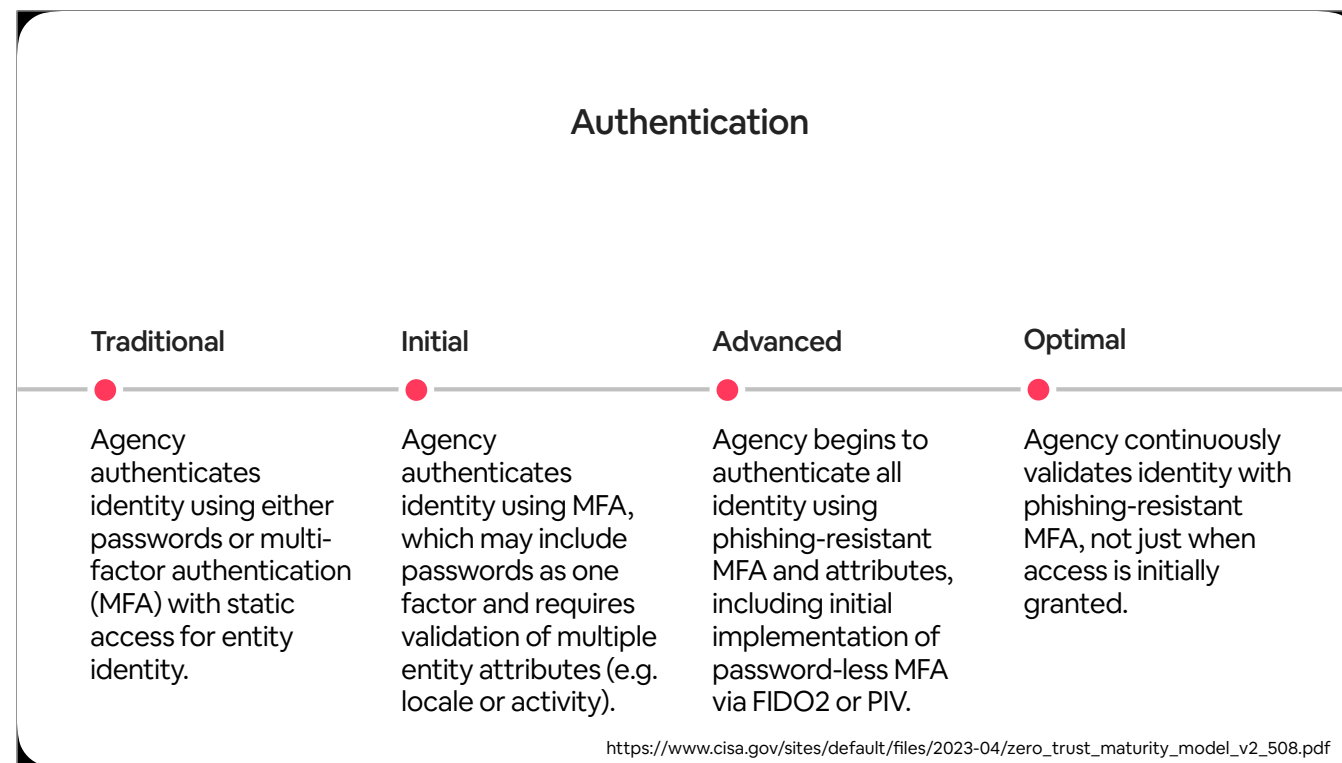


You start with a series of pillars - the CISA model has these pillars - they're a good starting point, but you may have other pillars you care about.



You will then gauge your level of maturity for each of those pillars from traditional, through to initial, advanced and then finally to optimal.

Generally these pillars will be too broad to gauge any real progress - you should split each one into smaller sub-pillars. For example with identity, you may split it into things like Authentication, Identity Stores, Access Management etc etc. Lets take a look at CISA's maturity model for authentication as an example



I'm not going to read this, but it illustrates how when you move through the maturity levels you incrementally improve your situation. When we decided on our own maturity model, we started with the CISA one, and then went through and made changes to reflect what we care about, where we are today and where we want to get to. Some of the items we were already Advanced or Optimal according to CISA, so we were able to push ourselves to be even better. Some of them we were definitely at Traditional, so we had work to do. The important thing here is to make sure you set appropriate goals for your organization. Unless you are a US government agency, you almost certainly will not care about all of the items listed by CISA.

Gap analysis

So you've listed your current tools and capabilities, you know what you want to achieve, and how you will measure success. You've even thought about some things that you want to do at some point in the future - it's time to identify your current gaps - where do your current tools and solutions fall short

When you've listed your current gaps, go through and at a very high level either identify things you have planned already that would solve the issue, or outline what it would take to solve the gap. If it's a project that's already planned, when are you doing it? If it's a new project, when might you do it? Lets take a look at a few projects you might want to consider

Device posture

Many popular MDMs offer an integration with your IDP for what they call device posture, but really most of them do a pretty poor job in my opinion. You should take time to think about what constitutes a healthy device in your environment - it might include the last time the device communicated with your mdm, checking that your security tools are running, making sure the device is on the latest version of macOS - there are lots of signals that you might want to consider, even just thinking about what it means to be a healthy, trusted device in your environment is likely going to be a project, let alone building out the pipelines to bring all of these data sources into whatever you are using for checking device posture and gating access.

Removing the trusted corporate network

What was unthinkable a few years ago now looks pretty realistic after we worked from home for three years. Write something about how just having a credential and being physically located in an office shouldn't give you special access, you need to assess the device posture and context as well

Policy decision point

You will likely need to think of what tool or service is going to act as your policy decision point - this is often the point where the combination of user, device and context are compared against your policy and makes the decision as to whether the access request is going to be allowed or not. This is where your data pipelines that send signals about the device will end up, so it is critical that you spend time on this

Continual authorization

It's all well and good validating the device posture when the user logs in for the first time in the morning, but one of the cornerstones of zero trust is continually assessing the device, user and context and making continual authorization decisions based on your policy - ideally it would be for every request, but you might want to at least think about a mechanism to make these decisions more regularly than just at login.

Identity verification

If you are doing things like enforcing the use of security keys or switching to password less, you will need pretty robust ways to verify the user's identity if they lose their device or access to their devices. This is a tough problem today when you have people who may have never met any of their co-workers in real life, and it is only going to get harder with things like deepfakes

IMPORTANT NOTICE

This one is very important - if the team that needs to solve the issue or be involved in the solution isn't present in the room, do not talk about it further. Identify who needs to be involved, and put it to one side until you talk to the team. Take improving your identity verification for example - it is likely that your help desk will be involved in fielding the calls for password resets, so they are major stakeholders in any solution, so absolutely must be involved in any project brain storming.

Mutual TLS for service to service communication

Here's another example of a project that you will need to have other stakeholders in the room for. As Client Engineers, you may not be responsible for all of the services that run in your data center or cloud of choice. Strongly authenticating service to service communication is an important step in a Zero Trust roadmap, but it is not one that we can take without the owners involved, who in my case would be our Systems Engineering team.

A large, light pink rounded rectangle with a thin black border. Inside the rectangle, the text "WOOOOAH we're half way there!" is centered in a bold, black, sans-serif font.

WOOOOAH we're half way there!

Indeed, we are half way through your planning session.

We've identified what we think our users want to do, we have identified what we have, what we want to achieve, and what our gaps are. It's time to wrap this up into a digestible format. You will probably have a ton of documents at this point. Let's make something you can share with the world, from leadership, to your partner teams, and perhaps even your end users.

The Four Pager

The goal of the four pager is to provide the right level of information to every level of your organization. The CEO could read the first page and have a good idea of why we are doing this, engineers can read the whole thing and get a good overview of the entire strategy

The Elevator Pitch

We, as engineers, know why we want to do zero trust, but we need a four sentence max statement on what problem we are trying to solve. It is critical that everyone on the project can deliver an “elevator pitch” - imagine you are in an elevator with your CEO and they ask you why you’re working on Zero Trust - you should be able to tell them during that elevator ride. This is ours:

Workers need a frictionless, secure way to access the resources they need. Our vision is a well-managed approach to access that protects Airbnb's assets and is a great experience for our users (Airbnb's workforce). Our goal is to provide the right level of access to the right identity on the right device with the right context.

I'm going to break my rules and read this slide, as it's important.

Workers need a frictionless, secure way to access the resources they need. Our vision is a well-managed approach to access that protects Airbnb's assets and is a great experience for our users (Airbnb's workforce). Our goal is to provide the right level of access to the right identity on the right device with the right context.

So this is our elevator pitch - we've solved the "CEO problem". Now we are moving down one level to our executives - why are we doing this?

Business Case

For that audience we are focusing on the business case - will this save time or money? If users are able to access resources without needing to connect to VPN for example, you're going to save time. Spend some time researching how long users take to auth to VPN every day. If VPN has less use cases, you might be able to reduce your license count for your VPN of choice. Those sort of things are important to include in addition to the security improvements

Outcomes

Let's take it a little lower - what we actually trying to do here - this should be an incredibly high level overview of what you're going to do. We focused on three pillars here - improving the user experience, improving security and reducing risk, and improving governance and oversight around how our data is used. Yours might be similar, they might be different - each organization has its own culture.

We've largely answered the "why are we doing this" question at this point. Now we are going to answer the "who" - who needs to be involved, who will be impacted

Target Audiences

Describe your primary groups of customers. Ours are full time employees, contractors, customer support agents and robots. You can take this opportunity to highlight one or two of their top pain points

User Journeys

This is where we are going to summarize our user journeys - you won't be going into much detail here, it's to give your stakeholders a sense of where users are right now and what they're doing. As with all of these sections, you can link out to more detailed documents if you need to.



Pain points

I believe that solving user problems is one of the most important parts of our job, and as such highlighting the problems we are going to solve for our customers is a vital part of your four pager. We split ours by mood - How annoying are these problems? Obviously we didn't have any in the amazing experience section YET - that's where we're going. But we could split them by slightly annoying - their name when they onboard for example is wrong, or they are having problems with a broken security key, and by REALLY annoying or bad - like then an OS update goes wrong and bricks their device or something.

The What

summarize your strategy - for us it was a table that summarized the maturity model and capability mapping document we spoke about earlier, which outlined everything we wanted to achieve. We aligned it with how it would improve our security posture and how it will improve the user experience.

The When

when are you doing this? When you identified your potential projects, you took a somewhat wild ass guess at when you were going to do things - pop them onto a timeline

How

Link out to further resources - the docs from your working sessions, jira dashboards, project pages, metrics dashboards, how you're going to measure success. You might not have much here yet, but you will be adding to this as you go - let your stakeholders know exactly what is happening

The End

Apart from the small matter of implementing everything

So that's how I think you should think about planning your zero trust strategy. I didn't come up with this alone, I was merely a contributor to the process - even the act of planning out how we were going to plan this thing took weeks and weeks. Before we close out, here's a quick recap

Recap

- ZT is not a product
- Keep your users in the front of your mind
- Make sure everyone on the project knows the elevator pitch
- ZT is a journey that impacts everyone

- Zero trust is not just a product you can buy. Hooking a conditional access product into your mdm is not going to really get you to a good place by itself - sure it might be better than what you have, but it is not really a strategy.
- Keep your users in the front of your mind. If what you provide isn't better than what they have today, they will work around it
- Make sure everyone on the project can describe accurately and succinctly what you're trying to achieve
- ZT is a journey that impacts everyone in an organization, not just security. Security teams cannot come up with this on their own. Subject matter experts from every area in IT and beyond need to be involved

Thank you!

grahamgilbert.com

graham.at/movember

@grahamgilbert@mastodon.social

graham.gilbert@airbnb.com

So that's it - you can find me on these places - I have a rarely updated website, I very occasionally post on mastodon and you can email me at my very cryptic and hard to guess email address. And if you found this talk useful in any way, please consider donating to the Movember foundation and help save some lives of people you might very well be sitting next to. And now we can take some questions