Zero Trust *Intelligence*: Open vs. Closed AI, Quantum Resilience and the Future of Digital Trust

Dan Laycock

In January this year at the World Economic Forum Annual Meeting in Davos, Elon Musk said that AI could become more intelligent than any individual human by the end of 2026, and more intelligent than humanity collectively by around 2030 or 2031. In a more recent interview with The Economist, he returned to the same theme, suggesting that AI could exceed the sum of human intelligence within around five years and that, looking towards 2036, its intelligence could be far greater than humanity's combined intelligence.

At the same time, another discussion is developing around how open that intelligence should be. Musk has recently said that X will open-source its entire codebase following a security review, including allowing independent reviewers to compare the published code with the software actually running on the platform. Mark Zuckerberg has spent several years making the case for open AI through Meta, while also acknowledging that the balance may change as models become more capable. More recently, Microsoft, NVIDIA, Meta, IBM and others have publicly backed the continued development of open-weight AI.

These conversations interest me. How powerful could AI become, and if we are going to place increasing amounts of intelligence inside the systems businesses and people depend on, how should we trust it?

I have been researching this alongside ideas I have been developing around quantum-resilient security and how SASE may need to evolve as the technology it protects changes. That work started with cryptography and a future threat we already know we need to prepare for. A sufficiently capable quantum computer could break some of the public-key cryptography used today to protect identity, data and communications. Post-quantum cryptography is now moving through standardisation and migration planning, with NIST standards already published and the UK's National Cyber Security Centre setting a migration path through to 2035.

AI presents a different challenge, but I think there is a point where the two begin to meet.

We are giving increasingly capable AI access to the identities, customer information, business data and systems we are trying to protect. AI is also becoming available to those trying to steal that identity, breach those systems and access that data. The architecture between the two has to protect against AI being used as part of an attack while allowing legitimate AI enough access and authority to be useful.

Quantum resilience does not solve that problem. It will not stop AI-generated phishing, prevent credential theft or tell us whether a model is safe. Its relevance sits underneath it - if identity, integrity, provenance and secure communication become more important as AI becomes more capable, then the mechanisms we use to establish those things need to remain secure as computing develops too.

I have started to realise that this is where my ideas and research into quantum-resilient security overlap with the current discussion around open AI.

Before getting into the security architecture, though, it is worth understanding what that discussion actually means.

What do we mean by open AI?

Open source is a familiar idea in software. The source code is available for others to inspect and, depending on the licence, use, modify and redistribute. It means we are not entirely dependent on what the business that developed the technology tells us about how it works. AI makes this more complicated because a modern AI model is not simply its source code.

During training, a model learns patterns and relationships from large amounts of data. That learning is represented through numerical parameters known as weights. Those weights are a significant part of what the trained model has become.

An open-weight model makes those trained parameters available. A business can potentially download the model, run it on infrastructure it controls, adapt it and build applications around it - that does not necessarily make it open source.

The weights may be available without the complete training data, training process, code or information needed to reproduce the model. The Open Source Initiative makes this distinction in its definition of open-source AI. Openness goes beyond access to the final weights and considers whether the system can genuinely be used, studied, modified and shared.

A closed or proprietary model takes a different approach. The developer retains control of the model and usually provides access through an application, platform or API. The customer can use the intelligence, but much of the underlying technology remains controlled by the provider. Ultimately it changes who controls the intelligence and where we place our trust.

Two sides of the same coin

There is a strong argument for openness.

If AI becomes a fundamental part of how businesses operate, allowing a small number of companies to control the most capable models creates dependency. Those companies can influence cost, access, functionality and the conditions under which other businesses build around their technology.

Open-weight models provide another option, giving businesses greater control over where models run and how they interact with their data, while creating opportunities for sovereign capability, independent research and development that is not entirely shaped by the priorities of the company that created them.

There is also a security argument.

Opening technology to wider scrutiny allows more people to examine it, find weaknesses and challenge claims made about it. Security research does not have to remain entirely within the business that built the technology. The challenge is that the same openness that distributes access also distributes control.

Once model weights are released, they can be copied and modified. Safeguards can potentially be removed, and models can be fine-tuned for purposes their original developers did not intend - if a serious problem emerges later, the original developer cannot simply recall every copy.

Closed models retain more of that control. The provider can update the system centrally, change safeguards and respond when vulnerabilities are discovered. The customer, however, has to accept more of what it cannot independently see.

Both approaches offer something valuable and introduce something uncomfortable.

Openness gives businesses greater control over the technology they use but gives the original developer less control over what happens to that technology after release. Closed gives the developer greater control but requires customers to place more trust in the developer. This is where I think the discussion becomes more interesting than open versus closed.

The underlying question is trust.

Where does trust sit?

To some degree businesses have, in most relationships, trusted the companies supplying their technology.

We assess the supplier, its security, reputation, financial position and track record. We look at certifications, contracts, technical capability and previous performance. The strength of the business behind the technology becomes part of our confidence in using it.

In simple terms:

Trust the business → Trust its technology.

AI makes that relationship more complicated.

A business may trust the company that developed a model without knowing exactly how that model will behave in every situation, particularly as it is updated, fine-tuned by others and exposed to data, applications and systems its original developer may never have anticipated.

As AI moves from answering questions towards taking actions, that difference matters more. An AI agent could access email, customer records and business systems, or initiate processes on behalf of a person without requiring their approval at every step, while increasingly interacting with other agents as part of that process.

Knowing who developed the model remains important, however - it is no longer enough.

We need to know what is acting, whether it is the version we approved, whether it has been altered, whose authority it is using and what that authority allows it to do.

That starts to reverse the relationship:

Verify the technology → Establish trust in the business.

This does not mean open technology is trustworthy because we can inspect it, nor does it mean closed technology is insecure because we cannot. It means trust increasingly needs evidence.

Protecting identity and data

This becomes much less theoretical when we think about what businesses, their customers and people actually need to protect. Identity and data sit at the centre of it.

AI can increase the speed and scale at which attackers operate. It can make impersonation and social engineering more convincing, assist with research and reconnaissance, and make some capabilities accessible to people who previously lacked the expertise to use them.

The weaknesses being targeted are often familiar, for example - credentials can still be stolen, access can still be misconfigured, and people can still be deceived. Sensitive information can still be exposed. AI changes the scale and sophistication with which those weaknesses can be targeted.

At the same time, businesses are giving AI legitimate access to exactly the things attackers are trying to reach. For AI to understand a business, it needs context. That may require access to documents, email, customer information, databases and applications. For agents to become genuinely useful, some will need authority to act on that information rather than simply read it.

The security problem therefore runs in both directions.

We need to protect people's identities and data from AI-assisted attack while allowing approved AI to work with those identities and data safely.

The security architecture sitting between the two was developed around people, devices, applications and networks. It has evolved repeatedly as the technology environment has changed. AI gives us another reason to evolve it.

From Zero Trust to Zero Trust Intelligence

Zero Trust emerged because location stopped being a reliable basis for trust. Being inside a corporate network did not mean a user was legitimate. A company-owned device could be compromised, and a valid account could be stolen. Applications and data had moved beyond the traditional perimeter anyway.

Security had to ask better questions about who was requesting access, what they were trying to access, what they were allowed to do and the context in which that request was being made. AI introduces a similar change.

We cannot assume that intelligence should be trusted because it was created by a business we trust, because it is running inside our environment or because it behaved correctly the last time we used it.

This is where I think **Zero Trust Intelligence** becomes worth exploring.

By Zero Trust Intelligence, I do not mean another version of Zero Trust Network Access. ZTNA applies Zero Trust principles to access. What I am exploring is whether those principles can be extended to the identity, integrity and authority of the intelligence increasingly acting inside our systems.

I am not presenting Zero Trust Intelligence as an established security architecture. It is an idea developing from research I have been doing around Zero Trust, SASE and quantum resilience, alongside the changes AI is beginning to introduce. The principle is not to make intelligence itself trustworthy, but to establish what we can about it and control the authority we give it.

Cryptographic signatures, attestation, identity and authorisation can provide evidence about where a model or software component came from, whether it has changed, the environment it is operating in and what an agent is permitted to do. None of this proves that a probabilistic model will behave correctly next time.

An AI agent should not receive broad access because a business has approved the model behind it. Its identity needs to be established, with access reflecting what it needs for the task and its authority reflecting the consequence of the actions it can take. Its behaviour can then be observed and that authority adjusted as the context changes.

Consider an AI agent used by a finance team.

It may need to read an invoice, compare it with a purchase order and prepare a payment, but that does not mean it needs authority to change a supplier's bank details or release the payment itself.

A person might approve the final transaction; a larger payment might require stronger approval. A change to the model, its environment or its behaviour might reduce its access. The question becomes more precise than whether we trust the AI.

Is this the approved agent, acting for the person or process it claims to represent, using the model we expect, under conditions we recognise, and should it be allowed to perform this particular action?

This isn't just theoretical. NIST is already examining identity and authorisation for AI agents, including how agents can be identified, authenticated and granted access when acting on behalf of people and business processes.

That does not give us Zero Trust Intelligence as a finished architecture. It does tell us that one of the problems underneath the idea is already real.

Where SASE might fit

SASE developed as the old network perimeter became less useful. Users, applications and data were no longer sitting together inside one environment. Networking and security controls had to follow them, with identity, policy and context becoming more important than physical location.

AI changes that environment again.

Access is no longer only about people and applications. An AI agent may now be acting on behalf of either, working across cloud platforms, applications and data environments, interacting with other agents or choosing different tools depending on what it is trying to achieve.

I don't think we can jump from that to saying SASE is the answer to AI security, but the question I am interested in is whether the policy, identity, data and access architecture developing around SASE could provide part of the enforcement layer we need.

If it can, the policy decision moves beyond whether a user can access an application to consider the intelligence acting on their behalf, the authority it carries, the information it needs and what it is attempting to do with it.

This is where the research I have been doing around Quantum-Resilient SASE starts to become more than a question of replacing cryptography.

Quantum resilience and the trust underneath AI

Quantum resilience needs to be understood for what it is. Post-quantum cryptography does not make AI safer. It does not stop an AI-enabled attack and it does not tell us whether an agent should be trusted. It can strengthen part of the trust infrastructure underneath those decisions against a different future threat.

If we need to establish that an agent is genuine, that it is acting for the person or process it claims to represent, that software or a model has not been changed, or that communications between systems have not been altered, cryptography can form part of the evidence. Identity, certificates, digital signatures, encryption and key establishment already underpin much of the digital trust we rely on.

Some of the public-key cryptography used for those functions today needs to change because of the future quantum threat. NIST has already standardised the first post-quantum cryptographic algorithms, and the UK's NCSC has set migration milestones running through 2035.

If we are already changing security architecture to accommodate AI, it makes little sense to build that future architecture around cryptographic assumptions we already know need to evolve.

This is where quantum resilience can strengthen the architecture we are developing. It does not solve AI security itself, but it does help make some of the mechanisms we depend on to establish identity, integrity and secure communication more resilient to future threats.

Crypto-agility matters for the same reason. Quantum resilience should not mean replacing one algorithm with another and assuming the work is finished. Businesses need the ability to change cryptographic mechanisms as standards, computing and threats develop. That is why I have been exploring Quantum-Resilient SASE as an architecture rather than a cryptographic upgrade.

The outcome should be understandable without needing to understand every technology underneath it.

Protect my identity and data. Know what is acting on my behalf and control what it can access and what it can do. Change that authority when the evidence or risk changes and make the trust mechanisms supporting those decisions resilient enough to evolve with the threats around them.

For a business and its customers, that is the value.

How close are we?

This is not a finished architecture today. Zero Trust and SASE are established. Post-quantum standards exist and migration planning is under way. Controls around identity, data, provenance, integrity and monitoring already exist in different forms. Work around AI-agent identity and authorisation is developing.

What we do not yet have is a mature architecture bringing those things together around increasingly autonomous intelligence. That is the area I think is worth exploring.

The challenge is not to invent security again from the beginning but to understand how the controls we already have, and those now being developed, need to work together.

An AI agent needs an identity, but we also need to understand who or what has delegated authority to it. Provenance and integrity need to give businesses better evidence about the models and systems entering their environments. Access controls need to recognise the difference between a person, an

application and an agent acting on behalf of either - the authority to access data should not automatically become the authority to act on it.

Underneath that, the cryptographic mechanisms supporting identity, integrity and secure communication need to evolve without forcing everything above them to be rebuilt each time the threat changes.

There is work to do, but there is enough happening now to make the direction worth pursuing.

So, open or closed?

This takes us back to where we started and I come down on the side of openness.

Not unlimited openness, and not because open technology is automatically safer. There will be capabilities where the potential consequence of releasing them changes the balance between openness and control - that risk needs to be understood rather than dismissed.

The reason I favour openness is that the alternative introduces a different risk.

As AI becomes more important, concentrating its most capable systems within a small number of companies risks creating a dependency where businesses, governments and customers increasingly rely on technology they cannot fully inspect, while the direction of innovation is shaped by those who control it.

Open models offer another path, giving businesses greater choice over where intelligence runs and how it interacts with their data, while opening the technology to wider scrutiny and reducing reliance on a small number of providers.

They also make misuse easier. That is the other side of the coin, and it cannot be ignored.

I don't think the answer is to make openness itself the source of trust. An open model should not be trusted because it is open, and a closed model should not be trusted because of the business behind it. The architecture should assume that neither is trusted simply by virtue of what it is.

Trust should instead be built around what we can establish about the intelligence, the authority we give it and the identity and data it can access, with that authority changing as the evidence and risk change.

This is why the current discussion around transparency interests me.

The important idea is not simply that code or weights can be made available. It is that claims made about technology can increasingly be tested against the technology itself.

That changes the relationship. For much of enterprise technology, trust has flowed in one direction:

Trust the business → Trust its technology.

AI creates the possibility, and perhaps the need, for another:

Verify the technology → Establish trust in the business.

That does not remove the need to trust businesses, but it gives us a stronger basis on which that trust can be built.

Elon Musk's prediction about where AI capability could be by the end of this decade gives us an indication of where this may be heading, bringing the debate over who controls that capability together with a much bigger question about how we protect the identity and data of the people, customers and businesses using it.

I think those questions eventually meet in the architecture, where quantum resilience can strengthen the cryptographic trust underneath increasingly capable AI, while Zero Trust Intelligence gives us a way to think about the authority we grant it. Openness adds another dimension by allowing that technology to be inspected and challenged beyond the claims of a single provider. The pieces already exist or are beginning to develop; the question now is how they fit together, where the gaps remain and how far that architecture can realistically evolve.

That is why I think this is worth researching now.

If we can get it right, the future of AI security does not have to depend on deciding which intelligence we trust, but on creating an architecture that establishes trust around it, controls its authority and protects the identity and data it can reach, while remaining capable of evolving as the technology and threats change.