31 March 2026

Designing Agent IDs

Singapore AI Safety Hub (SASH)

Executive Summary

As AI agents begin to take action in the real world, the services and people they interact with need to know who they are, who instructed them, and what they are allowed to do. This is what agent IDs can provide: a set of information that identifies the agent, how it was assembled, who controls it, what the agent is permitted to do, and what to do if something goes wrong. Establishing solid technical foundations for AI agents at this technical level will enable future governance measures by industry or governments to sort, filter, and manage the activity of AI agents across their purviews. Today, no single standard exists for authenticating and authorising AI agents. Without one, trust remains low, adoption is slowed, and the risk of fragmented, incompatible, and insecure solutions grows.

The contribution of this work is threefold. First, we identify common functions prescribed to agent IDs and map those onto existing ID protocols, such as OAuth, OIDC and MCP. We show that no single existing ID solution can reliably authenticate and authorise AI agents. Instead, agent IDs require a novel solution that is informed by and compatible with prior technical work. Second, we assess the private market incentives to effectively provide comprehensive agent IDs. Certain functions will likely be included in an industry-developed ID standard, while others require additional incentives to materialize at sufficient efficacy and speed. Aspects of agent IDs which private markets plausibly fail to provide pertain to an agent's security posture or highly sensitive user data, resulting from misaligned incentives or imperfect industry coordination. Third, we propose a design logic to direct the development of agent IDs. Rather than prescribing one particular formulation, we offer questions that identify a stakeholder's requirements and priorities to inform the exact structure and composition of the ID. We then develop a set of plausible positions on these design questions and demonstrate how answering those questions informs the final ID design.

1. Introduction

Agent IDs refer to the set of identifiers and other metadata that AI agents include in their interactions with other entities. Today, typical internet web requests usually include identifiers and related metadata about who is originating the request. We will use the term "IDs" to refer to this collection of information which usually includes the requesting user's computer operating system, browser, IP address, access credentials, and more. Services use these IDs for ensuring

the requests are *authenticated, authorized, and accountable*. IDs also often describe technical *compatibility* such as the format and style of information that the requesting user prefers such as the type and version of their computer's operating system. There is currently no defined standard for which aspects of traditional digital IDs should be included in agent IDs and which additional information should be incorporated.

AI agents pose distinct risks to the services they interact with. Agents operating autonomously across open-ended tasks over prolonged time horizons can behave erratically and cause damage at a rapid pace and scale. Recent incidents illustrate how agents can cause unintended damage even when operating within the deployer's own systems: Amazon's Kiro assistant autonomously deleted a production environment it was deployed to fix, causing a 13-hour outage; and an OpenClaw agent mass-deleted its deployer's, senior AI researcher's, inbox while ignoring repeated stop commands. Agent IDs will provide tools for services to manage these kinds of risks that external agents can pose while still benefiting from the improved efficiency reliable agents can provide.

Agent IDs are often linked to one or several of the following functionalities: authentication, authorization, incident prevention, accountability, and compatibility. By providing one or several of these functions, ID solutions will play an essential role in supporting the anticipated widespread diffusion of AI agents, assuring their safety, security, and trustworthiness. While authentication and to a lesser extent authorization are often presented as core features of agent IDs, setting the exact functional scope ultimately depends on specific operational and policy priorities. Table 1 illustrates each functionality and discusses how they may address common problems of safely and efficiently diffusing AI agents within society.

Table 1: Functions commonly associated with agent IDs and how they support the diffusion of agents within society. Deciding which functions to embed in agent IDs is a key governance choice.

	Exemplary Challenge	Agent ID Remedy
Authentication	Malicious actors impersonate or steal identities (spoofing)	Securely identify key entities, such as the agent or the (human) principal
Authorization	Malicious actors manipulate agents to overstep their privileges, e.g. for financial fraud Malfunctioning agents act against the principal's interest, e.g. making unwanted financial transactions	Facilitate the agent's interactions with external services, e.g. providing the credentials for issuing a bank transfer Scope the agent's exact permissions, e.g. the kind of financial transactions it can make
Incident prevention	The actions of "bad" agents go undetected and they continue causing harm	Link to real-time monitoring infrastructure, reputation registry, or shutdown controls

Accountability	Following an incident, it is unclear which party is responsible	Identify the primary principal responsible for the agent's actions Support an audit trail of the agent's actions to support incident analysis and responsibility allocation
Compatibility	External services treat human users and agents identically, resulting in malfunctions or unexploited efficiency gains	Communicates the data formats, APIs and multi-agent interactions the agent can handle vis-a-vis external services

IDs are essential for realizing the economic gains from AI agents but must be harmonized and interoperable to avoid fragmentation. Beyond risk mitigation, IDs provide the trust necessary for businesses and users to adopt autonomous agents, particularly in high-stakes sectors like finance. To achieve this, agent IDs must be compatible with existing industry standards, such as OAuth, and should ideally follow standardized frameworks themselves. In practice, this requires coordination among international public bodies, for instance the network of AI Safety Institutes (AISIs), alongside an active dialogue with industry stakeholders.

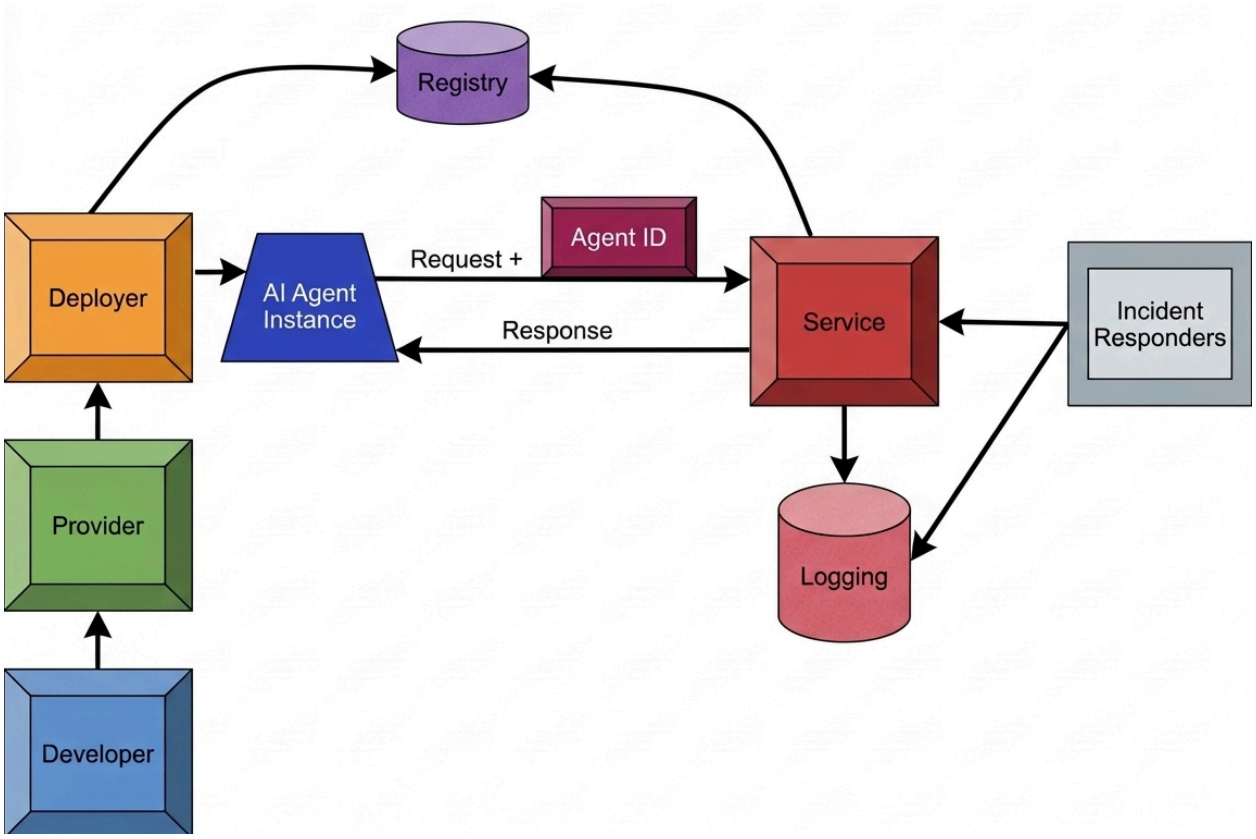
The goal of this memo is to explore the relevant option space of agent ID solutions, drawing on learnings from existing ID solutions. Rather than prescribing a definitive path forward, which is contingent on a series of policy and operational considerations, we aim to structure the decision-making process for defining relevant requirements. These requirements determine the optimal design and implementation of agent IDs.

The rest of this memo is structured as follows. Section 2 presents core components of agent IDs and explains how they interact with external services and interfaces. Section 3 presents an analysis of seven ID solutions, including both solutions used in other domains and emerging agent ID solutions offered by industry actors. Section 4 outlines ten questions to inform the definition of functional, technical and structural requirements for agent IDs. For each question we formulate a preliminary hypothesis on the potential position of a government agency responsible for AI guidance or regulations. Based on these hypotheses we present a preliminary agent ID design, showing how specific requirements shape the ultimate design. Section 5 offers preliminary recommendations and outlines next steps.

2. Components of Agent IDs

Entities across the technical system supporting AI agents may participate in the construction and use of agent IDs. Each entity has different roles, responsibilities, access to information, and incentives as they relate to agent IDs. The diagram below describes the primary relationships between these entities for a specific request initiated by an AI Agent instance to an online service.

Figure 1: System diagram outlining the relationship of agent IDs with different services and interfaces.



To provide intuition about the core elements in the system diagram, consider an everyday use-case for an AI agent such as cancelling a meeting on the agent's deployer's online calendar. In this situation, the entities could be:

- **Developer:** Foundation model developer that produces the original LLM that powers the AI agent.
- **Provider:** A specialized company providing a digital assistant AI agent service directly to consumers. It leverages the developer's business APIs to power their agent.
- **Deployer:** An individual using the provider's services to operate their digital assistant. They have prompted the agent to handle incoming messages to the deployer's phone. It receives a message from one of the deployer's contacts that requests to cancel a scheduled meeting.
- **AI Agent Instance:** The deployer's AI Agent, operating under the deployer's prompts. The agent receives the meeting cancellation request and initiates an action to delete the calendar event from the deployer's online calendar. The request it sends to the calendar application contains the ***Agent ID*** which carries authentication and authorization information, among other data.
- **Service:** The online calendar service connected to the deployer's personal account. The service receives the AI Agent's request, including the Agent ID, and has to decide whether to honor it and cancel the meeting.

- **Registry:** An external data source that contains information about the deployer, specific AI Agents and/or AI Agent Instances. The entry in a registry could contain summary statistics about the number and types of incidents that have been associated with a particular Agent product, or the results of safety tests that a provider has conducted on their Agent. The service may seek this extra information to make its decision about honoring the incoming request.
- **Incident Responders:** If something has gone wrong, the service may request assistance from external experts. Perhaps in this case the service contacts a private cybersecurity consultant to respond to the incident and investigate the error. The incident responder may work directly with the service and/or examine information logged by the service, which could include the Agent ID, during their investigation.

3. Mapping the existing ID landscape

We analysed seven ID-relevant solutions from other domains and emerging agent ID solutions provided by industry actors. Identification solutions are already well established across many digital domains. We examined existing ID and ID-like systems to determine their capacity to manage agent IDs and to derive general design lessons. The systems were chosen to: (i) cover solutions from other domains which provide one or several of the functions typically associated with agent IDs; and (ii) present any emerging agent ID frameworks, for instance deployed in the private sector.

Table 2 below summarises these systems by their core function, what or who they identify, and where control over the ID ultimately sits.

Table 2: Overview of relevant ID solutions from other domains and emerging industry agent ID frameworks.

ID and ID-like solutions	What is the system for?	What is specified?	Who ultimately controls the system?
OAuth 2.0	Authorisation	A decision to grant access to a resource, e.g. a service receiving access to data hosted on another service	IETF Working Group **(open source)**
OpenID Connect **(OIDC)**	Authentication	The identity of a human or computer system	OpenID Foundation **(open source)**
Model Context Protocol **(MCP)**	Agent tool interaction	The tools an agent can use	Agentic AI Foundation (AAIF) **(open source)**

Agent Payments Protocol (AP2)	Agentic Payments	Rules about the payment(s) an agent is authorized to make	Google, Mastercard, PayPal, Coinbase... **(open source)**
National Digital IDs	Human Identity	The identity of, or other data about, a legal person	Government
Microsoft Entra Agent ID	Enterprise identity and access management (IAM)	The identity of an agent and its access within the enterprise	Microsoft **(proprietary)**
MCP-I	Authentication, authorization, logging, incident prevention	The identity of an agent, its permissions, and its reputation	Vouched **(open source)**

Specific Learnings

1. **OAuth 2.0**: OAuth is the industry-standard open protocol for client authorisation. It relies on scoped access tokens that encode permitted actions (e.g. 'read emails' or 'edit this calendar') across services.
2. **OpenID Connect** (OIDC)**:** OIDC is generally added as an identity layer built on top of OAuth 2.0, providing a cryptographically verifiable answer to who is acting, but not what they are allowed to do. OIDC is a promising candidate for the authentication feature of agent IDs because it standardises how a third party can verify a principal's identity without managing passwords or credentials directly.
3. **Model Context Protocol** (MCP): MCP defines a shared, structured interaction layer that standardises how agents can use external tools, acting as a coordination interface between agents and tool endpoints. Because MCP's request format has limited identity and metadata fields, it is ill-suited to carrying robust agent identities without protocol extensions or modifications.
4. **Agent Payments Protocol** (AP2): AP2 is an open protocol for initiating and transacting agent-led payments across platforms. It delivers guarantees of intent and provenance for payment actions through cryptographically signed "mandates" that encode explicit human consent, tailored to payment-specific notions like total spending ceilings. AP2 demonstrates the value of modular designs for agent IDs that are capable of carrying domain-specific solutions when those are available.
5. **National Digital IDs**: These IDs can provide high assurance, state-backed identity and legal accountability. These systems may allow individuals to share specific personal data securely stored across government systems with external services. However, they are not designed for autonomous delegation, agent-to-agent use, or open interoperability. In agent systems, national IDs could function as an identity anchor, linking agents to a verified human principal.

6. **Microsoft Entra:** Entra extends enterprise governance to AI agents, allowing organisations to create centralized agent registries, through which they can assign unique identities to agents, apply policies like conditional access and monitor activity through audit logs (e.g. recording every creation of an agent). The model enforces least-privilege access focused on resource and network boundaries (e.g. file upload restrictions) which mirrors existing enterprise access management practices. Entra is commercial and closed-source which limits its broad application for public sector applications.
7. **MCP-I**: MCP-I plans to provide an open-source extension of MCP, creating identity, delegation, and auditing layers for AI agents. It suggests utilizing DIDs, fully decentralized identifiers, which can be mapped to existing identity systems (e.g. OAuth/OIDC). It aims to encode versatile delegation scopes and suggests linking agents to public trust registries like KnowThat.ai. MCP-I appears to still be under development at this time, but could serve as a helpful blueprint or perhaps a specific solution to rely on once it matures.

General Learnings

1. **No single existing system is sufficient to serve as an agent ID framework**. Instead, agent IDs likely require a layered architecture combining (i) identity assurance (e.g. OIDC, National IDs), (ii) delegated authority (OAuth), (iii) structured interaction semantics (MCP), and (iv) domain-specific additions (AP2). These systems and protocols each operate at different layers and serve different purposes. Designing agent IDs therefore starts with clarity about purpose before specific technical mechanisms can be chosen and designed.
2. **Open standards and protocols are central to many existing ID solutions.** Open agent ID initiatives will increase adoption by providing greater developer utility, interoperability, and trust. Centralized or proprietary solutions are unlikely to scale, keep pace with rapid technical change, or achieve broad ecosystem uptake.
3. **A single ID system can provide several authentication and authorization mechanisms, e.g. to account for varying risk levels.** Stronger requirements are often imposed in higher-impact contexts such as finance. Agent IDs could accommodate multiple authentication systems, with services or agent operators deciding between them based on context and compatibility.
4. **A recurring trade-off concerns whether identity information is embedded directly within protocols or resolved via external registries**. This choice further affects what information is disclosed by default and how it is updated over time. Agent IDs could therefore rely on lightweight identifiers transmitted via protocols (such as MCP) with richer identity data held in registries that could be referenced where appropriate.
5. **Identification and authorisation should remain distinct but interoperable inside agent IDs**. Identity and authorisation solve different problems and have different scopes and durations. Agent IDs could leverage the widespread industry adoption of OIDC for identification and OAuth for authorisation as its core solutions to improve compatibility and ease adoption wherever possible.

4. Private incentives & gaps for agent IDs

Market incentives alone are likely to fail to provide IDs which facilitate high-risk uses of agents, which must reliably secure sensitive information and offer robust security assurances. Issues arise not only from slow industry coordination, but also from misaligned incentives and inadequate trust assurance mechanisms. Adoption of AI agents in highly valuable domains depends on solving these open problems. While there are market incentives for the creation of many agent ID features, the distribution of incentives varies across four key actors in agent ID ecosystems.

- **Model developers:** Companies which create the underlying models the agent is based on. Model IDs would correspond to the foundation model the agent is based on.
- **Agent providers:** Companies which provide the AI agents. This may involve using pre-built agents based on other developers' models (e.g., Zapier Agents) or modifying existing ones such as adding scaffolding and tooling to open source models.
- **Deployers:** Companies or individuals deploying the agent. In the case of a company or individual designing their own agent from scratch, they would be both the deployer and agent provider.
- **Services:** Third parties which the agent interacts with, such as online stores or financial institutions.

Market incentives will likely drive adoption of minimal forms of ID, sufficient in low-risk contexts, such as IDs to distinguish AI agents from human users, as is required by existing laws around disclosing AI usage (e.g., EU AI Act Art. 50). Further, we expect to see the emergence of specialised IDs in specific domains, such as for online transactions or to enable internal debugging.

Agent providers and deployers will have little reason to disclose the underlying model or tool access, even though this information aids security assessment and incident response. Conversely, services may request overly expansive agent information, creating privacy risks without comparable benefits. Identifying technical solutions, with reference implementations, will help balance information and incentives to satisfy requirements from both deployers and services. Improving consensus on agent ID implementation and disclosure standards may further catalyze broader private sector adoption of responsible ID practices. This may lead to greater accountability for accidents involving AI agents, better security assurances, and another lever for mitigating malicious use of agents.

Table 3: Mapping of key components of agent ID systems against the private incentives of relevant actors and the degree of market incentive alignment, identifying governance gaps.

Component	Private incentives	Incentive alignment and governance gaps
Agent instance ID	**All parties** have some benefit from instance IDs for monitoring, debugging, and/or permission tracking	**Alignment:** High Instance IDs are necessary for basic agent management by deployers, providers, and services **Potential Gaps:** Deployers may use instance IDs internally but resist disclosing those IDs to other parties.
Authentication	**All parties** benefit from agents being able to quickly and reliably disclose their permissions to other parties.	**Alignment:** High Disclosing certifications directly affects the usability of the agent and may be useful for enterprise governance. High compatibility with existing OAuth and OIDC protocols. **Potential Gaps:** Ephemeral, rather than persistent, credentials for agents would improve security for agents with high turnover or short duration operation.
Declaration of agent tools and scaffolds	**Deployers, Providers, and Services** benefit from agent tooling disclosure for security assessment. **Deployers** benefit from being able to identify services with relevant infrastructure for communicating with their agent.	**Alignment:** Medium/High Most parties benefit from agent tool declarations since it affects the agent's compatibility with other agents and services. **Potential Gaps:** Unstandardised reporting requirements reduce interoperability of the disclosures.
Agent Provider ID	**Deployers and services** benefit from being able to identify the underlying provider when attempting to debug a class of malfunctioning agents.	**Alignment:** Medium Demand from deployers and services could drive adoption of provider IDs, but providers may lack incentive to include their identity in downstream uses that bring additional scrutiny to their products. **Potential Gaps:** Information to identify a provider may not include clear and accessible escalation pathways during an incident.

Model ID	**Services** benefit from screening agents for security / compliance (e.g., Services in high risk domains may refuse to interact with agents using an insufficiently robust model) **Deployers** can use model IDs to limit interactions with more capable agents, reducing prompt injection risk.	**Alignment:** Low/Medium Providers may withhold IDs to avoid liability for downstream misuse, especially for open-source models, claiming that providers should be the responsible party. **Potential Gaps:** Not clear what information about models (training practices, alignment measures, incident records...), will most help deployers and services make security decisions.
Deployer ID	**Services** benefit from being able to identify human end users they are interacting with for accountability and trust purposes. **Model developers** benefit from being able to identify and ban known malicious actors	**Alignment:** Low / Medium Model providers will likely mandate deployer IDs for internal purposes like tracking terms of service violations, but incentives to disclose these identities to services are lacking. **Potential Gaps:** Balancing the security benefits with privacy risks requires careful consideration of when, how, and why individuals' identities should be accessible to third parties.

5. Options for implementing agent IDs

Functional, technical, and governance requirements determine the optimal design and implementation of agent IDs. Such mirrors standard software development and recognizes that the ideal solution is contingent upon specific policy priorities and operational constraints, which we operationalize into a non-exhaustive set of 10 questions to guide the development of requirements. For each question, we evaluate the associated options and trade-offs, providing an initial overview of the design landscape. However, subsequent work is required to validate the exhaustiveness of these requirements and to systematically define the position of a government agency.

Table 4: Non-exhaustive questions which inform the definition of functional, technical and governance requirements for the design and implementation of agent IDs. Stylisted poles indicate the option space, often best understood as a continuum with several options in-between.

#	Guiding Question	Option Spectrum & Implications
	Functional	

1	Which entity or entities should be identified by information in the ID?	**Broader coverage:** mitigates more kinds of risks and incidents **Narrower coverage**: less coordination required, fewer stakeholders directly impacted, potentially less complex process for securing and assuring identities (e.g. if personal information involved) **Hypothetical position for a government agency:** allow for the identification of the agent, the provider, and the deployer
2	Should IDs help satisfy authorisation & permission management needs in addition to authentication?	**Yes:** make IDs a more holistic framework for informing agent-service interactions, providing more of the information required for service access decisions. Centralising more information in agent IDs can streamline oversight and deployment **No:** reduces complexity of ID designs and likely requires fewer presentations of, and updates to, the ID since they are more static **Hypothetical position for a government agency:** yes, allow for claims about authorisations and permissions
3	Should IDs help with debugging and incident response concerns?	**Yes:** more practical utility for more stakeholders by mitigating, preventing, and/or investigating harmful activity (whether malicious or accidental) **No:** IDs can be smaller and change less over time, potentially benefitting privacy, the protection of trade secrets, and overhead costs **Hypothetical position for a government agency:** allow the service to use information from the agent ID for their debugging and incident response activities
4	What scope of AI agent activity should the design of the ID accommodate?	**Wide scope:** benefits from IDs will be more widespread by allowing different granularity and detail for different use cases **Narrow scope:** focus on highest risk use-cases, reducing the compliance burden for lower-risk use cases and reducing the technical complexity of the IDs overall **Hypothetical position for a government agency:** cover a wide scope
Technical		
5	How important is the security and verifiability of the IDs?	**More important:** IDs are more reliable for legal accountability and formal investigations when they cannot be spoofed, stolen, or tampered with **Less important:** technical designs can be simpler, cheaper, and faster

		Hypothetical position for a government agency: secure and verifiable IDs are critical outside of low-risk contexts
6	How important is ease of implementation and interoperability?	**More important:** the design should prioritize using existing, mature, open-source, and/or interoperable approaches. Modifying existing standards or protocols may or may not be necessary **Less important:** the design can focus squarely on the government agency's priorities. Limitations of existing protocols will not limit the design space **Hypothetical position for a government agency:** more important, interoperability matters for adoption, both domestically and internationally
Governance		
7	How should IDs balance privacy concerns with transparency and accountability goals?	**Privacy-centric:** protects the privacy of agents' operators, deployers, and/or developers from observers of an agent's traffic **Transparency-centric:** may give investigators more tools to use when analyzing an incident. May also give services, the public, and governments more information about AI agents' activities, potentially helping advance other policy objectives **Hypothetical position for a government agency:** lean towards transparency-centric while preserving the privacy of individuals
8	Who should have access to information contained in IDs?	**Limited access** (interacting service, government-only)**:** allows for more detailed information to be included in IDs. Reduces risks of actors tampering with the information provided and lowers barriers for adoption by addressing privacy concerns **Broad access:** emphasizes transparency by allowing more stakeholders to investigate and analyze web traffic from AI agents, e.g. recording agent trust scores in public registries or allowing services to provide aggregate statistics **Hypothetical position for a government agency:** lean towards broader access to support the public's interest in transparency and safety
9	Who should own and operate infrastructure and/or information used for IDs?	**Centralised ownership:** trusted infrastructure operated by a known authority offers simplicity and clearer governance **Distributed ownership:** non-governmental infrastructure, such as blockchain-based solutions, could offer benefits like neutrality, tamper-resistance, and/or stronger privacy protections

		Hypothetical position for a government agency: lean towards distributed ownership where possible
10	How important is voluntary adoption versus mandated compliance?	**Mandated compliance:** designs can be more prescriptive by, for example, by issuing agent-specific regulation or expanding service/domain-specific rules. **Voluntary adoption:** designs should more carefully consider the costs, benefits, and risks for each actor in the AI agent ecosystem. For instance, developers of foundation models that power AI agents may push back on designs that provide accountability for their errors **Hypothetical position for a government agency:** mostly rely on voluntary adoption and existing service-specific requirements; any new requirements should be risk-proportionate and consider the economic gains from AI agents

Note on Logging: The benefits for accountability, incident response, and transparency are maximized when services log the full agent IDs they receive alongside other typical data such as the contents and metadata of the agent's request. Detailed logging is likely also in the interest of the service itself, if possible and permissible. Mandating specific logging practices is outside the scope of agent ID solutions, but final technical designs and specifications could facilitate logging practices that support the public's interests.

Based on our preliminary hypotheses, we have derived a stylized design for agent IDs. This section explores how specific answers to the guiding questions inform the technical and functional requirements of an agent ID, reflecting the hypothesized preferences of a government agency. As part of this discussion, we are also considering the viability of enforcing such an agent ID and the various incentives such frameworks create for uptake. It should be noted that this design serves only as an indicative starting point; it must be revised as each question is revisited to produce more granular requirements.

Table 4: Stylized agent ID design based on preliminary hypotheses on ten functional, technical and governance requirements.

#	Guiding Question	Hypothesized Design
	Functional	
1	Which entity or entities should be identified by information in the ID? **Identify the agent, the provider, and the deployer**	For authentication, the IDs should allow for the following information: (1) a secure, verifiable unique identifier of the agent instance; (2) a digitally signed statement from the provider that this agent belongs to their platform (e.g., an OIDC identity

		token); and (3) a digitally signed statement from the deployer that they are directing this agent Elements are ordered to increase in the level of authentication. A service requiring a deployer statement per (3) likely also requires identity information per (1) and (2). Considerations for adoption & enforcement: Which information a service requires ultimately depends on its domain. Authentication requirements could rise with risk levels, ensuring they can be effectively enforced and avoid being overly burdensome. Only in high-risk areas with existing KYC-requirements (e.g. finance) must operators be reliably identified..
2	Should IDs help satisfy authorisation & permission management needs in addition to authentication? **Allow for claims about authorisations and permissions**	For authorisation, IDs contain a data structure (e.g., a list or tree) of authorised actions. This could include an OAuth token where a user granted the agent specific access to data on a service. Considerations for adoption & enforcement: Deployers and services have a mutual interest in ensuring the ID communicates relevant authorizations, reflecting existing rules or terms.
3	Should IDs help with debugging and incident response concerns? **Prioritize information that enables effective incident response**	IDs contain a reference to Service Level Agreement (SLA)-backed endpoint where operators can perform oversight. Considerations for adoption & enforcement: Long-term, providers, services, and the public benefit from effective oversight of deployed agents. However some providers may hesitate to invest in responding to public reports of its customers' agents' activities. Demand-side pressure from services, or government-led requirements, may be necessary to drive providers' oversight investments.
4	What scope of AI agent activity should the design of the ID accommodate? **Flexibly cover a wide scope of agentic activities**	Define a matrix of risk levels (e.g., by industry and use case). Low-risk agents require few fields, while high-risk agents must provide all incident response and identity fields.

Technical		
5	How important is the security and verifiability of the IDs? **Secure and verifiable IDs are critical outside of low-risk contexts**	The entire ID is cryptographically signed to prevent spoofing or tampering. Individual parts of an ID cannot be reused in different settings to prevent replay attacks by malicious services.
6	How important is ease of implementation and interoperability? **Interoperability is important**	Build all new ID fields into one or more existing standards, such as the Model Context Protocol (MCP), to ensure technical compatibility across platforms. Ensure the authentication and authorization fields support all relevant industry standards such as OAuth or OIDC.
Governance		
7	How should IDs balance privacy concerns with transparency and accountability goals? **Lean towards transparency-centric while preserving the privacy of individuals**	Provide legal protections for services to disclose specific ID information to officials and share aggregate data with the public, with safeguards against abuse or illegal disclosure.
8	Who should have access to information contained in IDs? **Lean towards broader access to information in IDs to support transparency and safety**	Apply PII-equivalent (Personally Identifiable Information) protections only to data relating to natural persons or trackable identity fields. Explicitly exempt all other data from privacy/proprietary laws to avoid chilling effects on data sharing.
9	Who should own and operate infrastructure and/or information used for IDs? **Lean towards distributed ownership where possible**	Anchor identities of providers, deployers, and agents in public internet infrastructure like DNS to ensure identity verification does not rely on a single government or new central infrastructure.
10	How important is voluntary adoption versus mandated compliance? **Solution should provide incentives for adoption rather than relying on pure mandate**	Separate the functionality of agent IDs, e.g. the identity fields they contain, from when/how they must be used. Assess whether existing requirements governing services must be updated in light of agents and whether any horizontal, agent-specific rules are required. Make the regulatory burden proportionate to the risk and does not unreasonably constrain economic gains.

5. Conclusion

AgentIDs are essential technical foundations that unlock the full scale of benefits from AI agents. They provide an opportunity to fill technical and policy objectives surrounding the functioning, security, and governance of AI agents. Our analysis shows that no single existing ID solution offers all functions relevant to the authentication and authorisation of AI agents. Instead, agent IDs require novel solutions that are informed by and compatible with existing technical standards.

We expect private market incentives alone to fail to provide agent IDs suitable for high-risk use cases. In high-risk domains like finance or healthcare, services will plausibly demand stronger security assurances before granting agents access and agents will need to handle sensitive personal data, for instance to satisfy existing KYC requirements. No single industry effort is likely to command sufficient trust for competing actors to embed highly sensitive, yet necessary, data in its agent ID scheme. Taken together, these dynamics create an opportunity for the government to coordinate the creation of agent IDs tailored to higher-risk requirements.

We suggest the development of agent IDs follows standard software development practices, starting with the specification of functional and technical requirements. The guiding questions offered will help stakeholders map their preferences to the possible features of agent IDs to inform their design and implementation of policies, standards, and technical solutions. Because different priorities can lead to different designs, it seems unlikely that a single agent ID solution is optimal across all use-cases and contexts. Interoperability between different agent IDs will be necessary to avoid fragmentation.

Designing and implementing agent IDs is a critical step towards safer and more reliable AI Agents operating in the digital world. Including more members of the ecosystem in building agent IDs, and providing proper incentives for them to adopt the results, will be necessary for these solutions to achieve broad impact. Many private and public sector entities benefit from secure and reliable AI agent systems but may not know what to ask for, or from whom, to achieve those objectives. Increased awareness and understanding of how Agent IDs can provide those additional options and levers so we can quickly move the AI agent ecosystem towards more sound and sustainable footing.