

AGNTCon + MCPCCon · OCTOBER 23, 2026

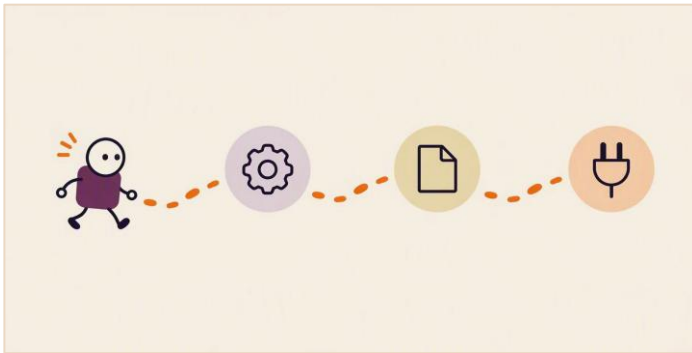
Agent Governance Lives in the OS

Roberth Karman
Alexander Sklar

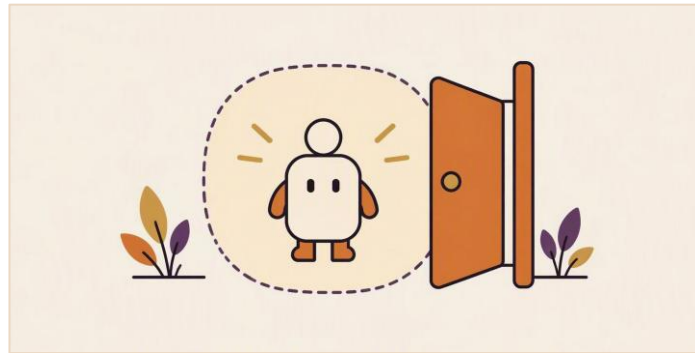


Governing Agents

What did it do?



What was allowed?



What could it not do?



Who authorized that boundary?



Governing Agents

INSTRUCTION

“Don’t read SSH keys”



Agent complies

SELF-SELECTED

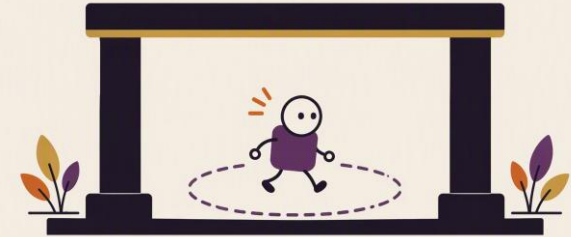
Picks working scope



Agent chooses scope

ADMIN-GOVERNED

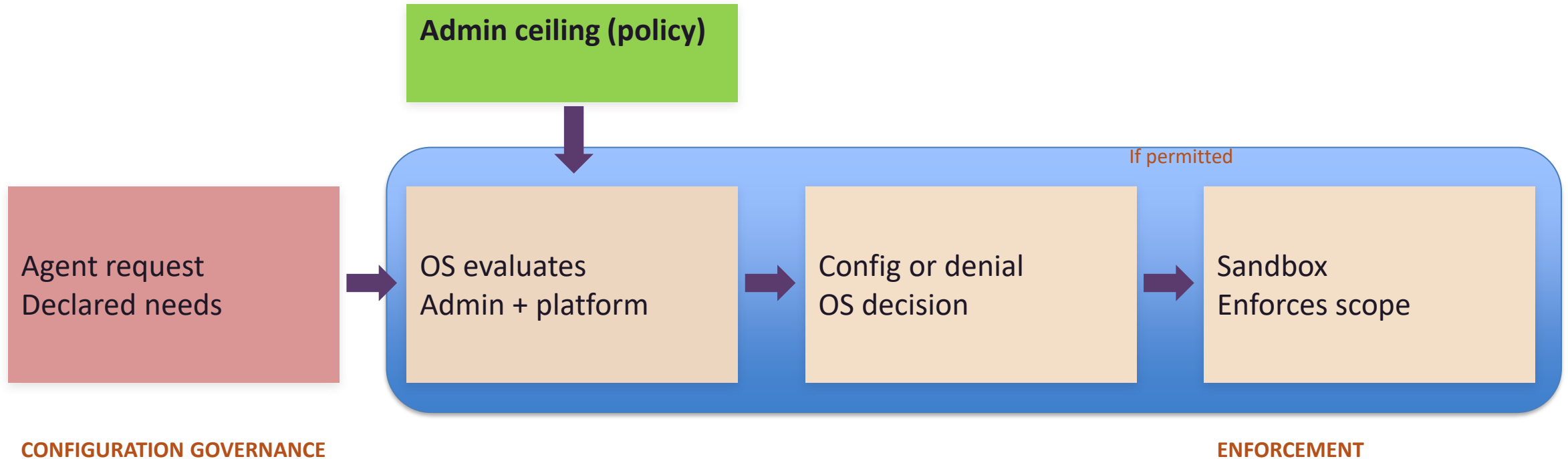
“Admin sets the ceiling”



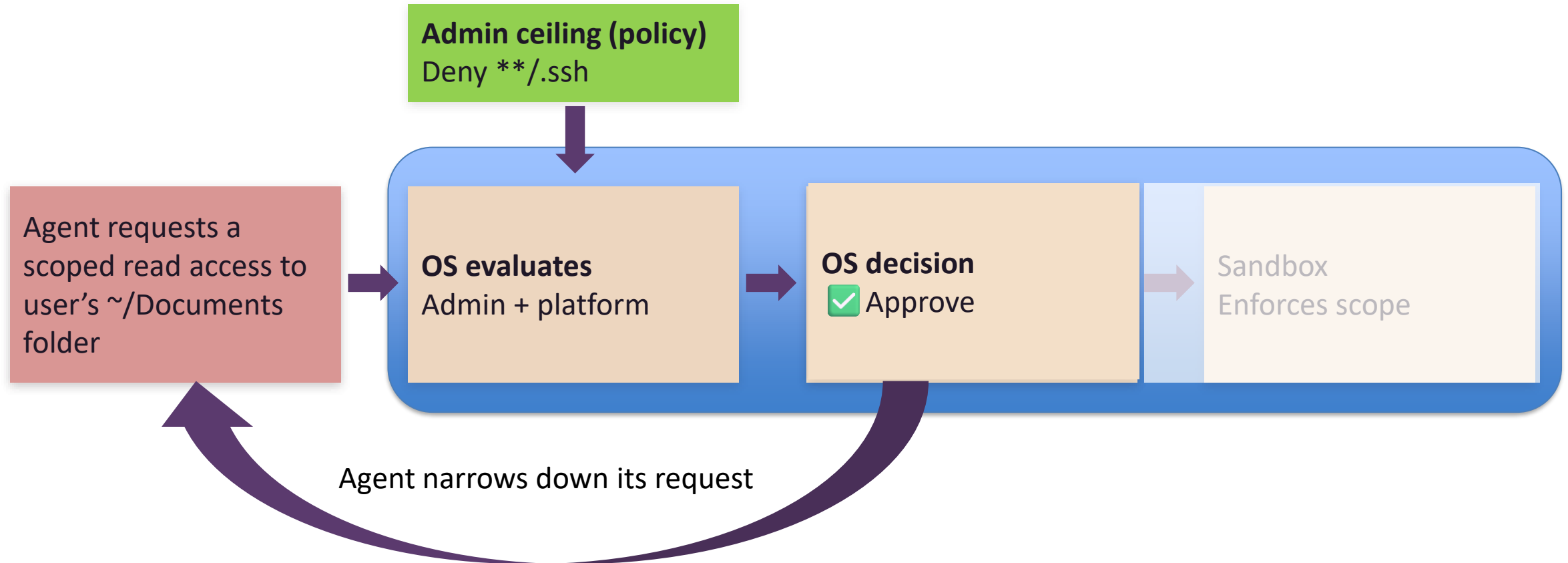
OS holds ceiling

Enforcing a boundary $\neq$ authorizing its scope

Runtime requests. OS decides. Sandbox enforces.



Asks for more? Enforce the ceiling.



Sandbox principles



Sandboxes are governed

- Administrator policy is always applied
- Sandbox requestors have identity



Sandbox permissions are immutable



Sandbox permissions cannot broaden

- Sandboxed processes need to request a higher-privilege process to do so (“rendez-vous”)

Same action. Same policy goal.

Sandboxed CLI

```
python summarize.py
```

Workspace allowed.
Credential excluded.

Agent's own process

Direct resource access

In-process read.
Policy mediation needed.

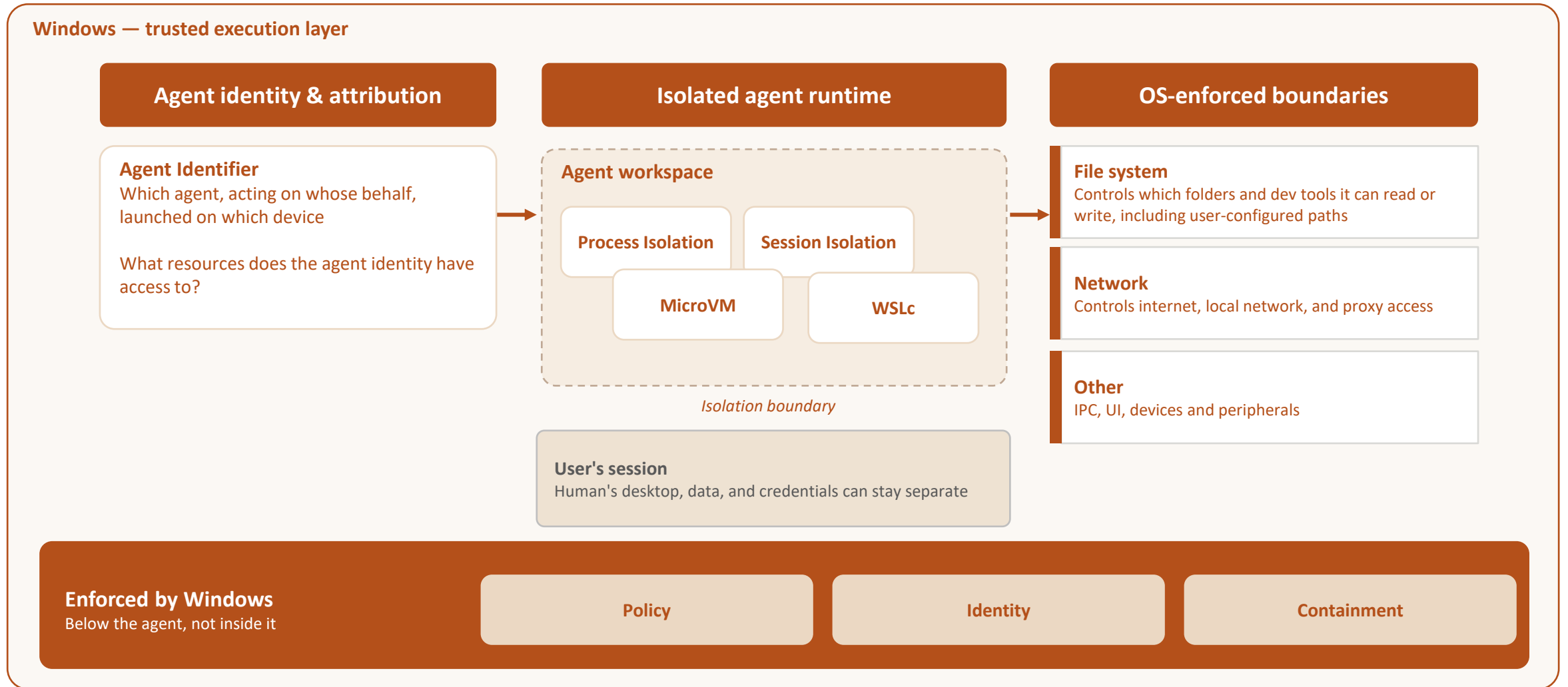
Helper or service

Delegated resource access

Delegated read.
No borrowed authority.

The execution route must not decide whether policy applies.

The OS as a trusted AI Gateway



Adopt the governed future



OS vendors

Enforce sandbox ceilings; extend action mediation



AI Gateway vendors

Leverage OS primitives for containment and governance



Agent vendors

Leverage sandboxes, request least scope; preserve policy across routes



Tool vendors

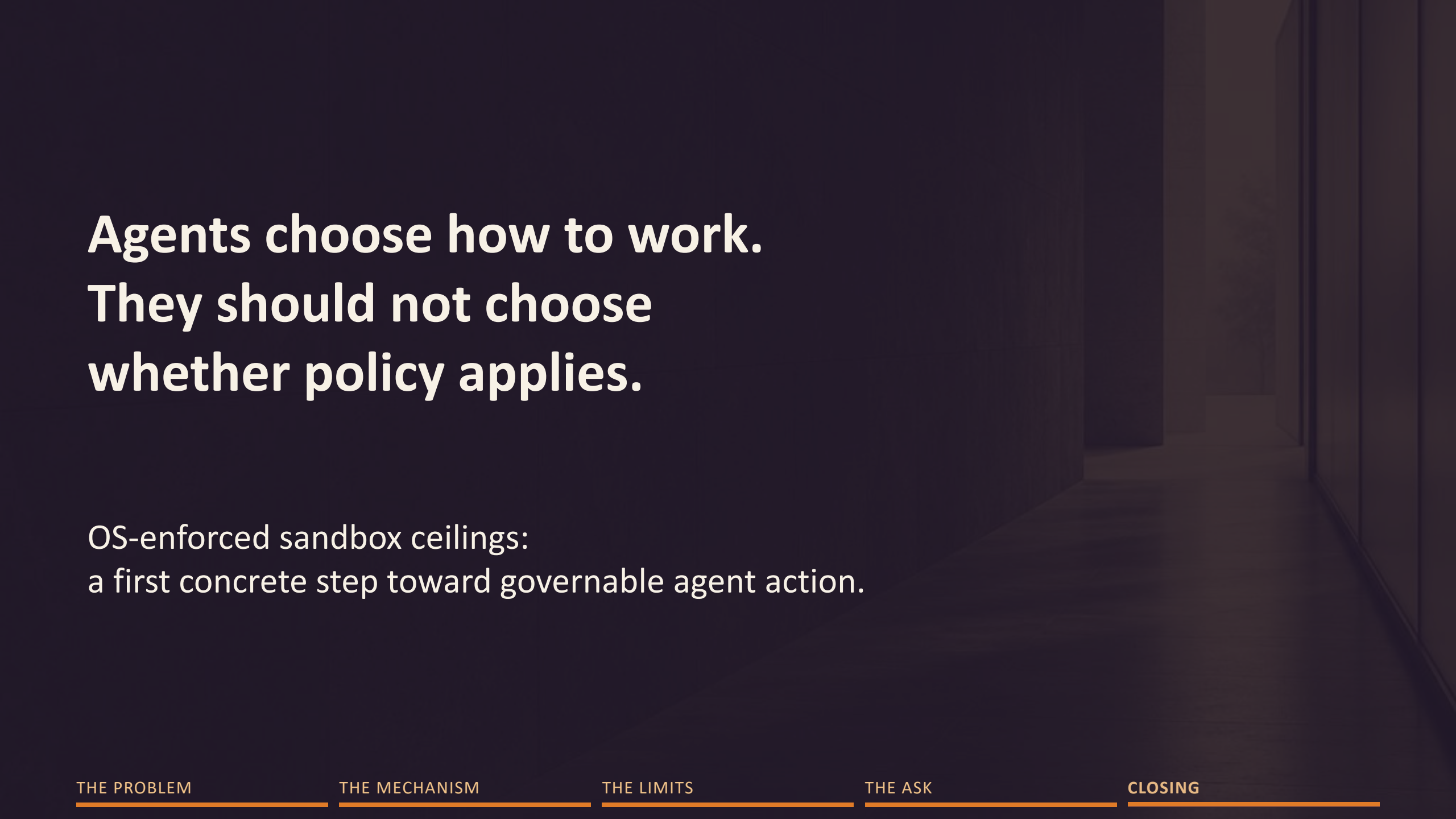
Declare minimal needs



Admins

Inventory your organization's usage; Define policy

Use enforced paths wherever they exist, and make the boundaries hold wherever they do not.



**Agents choose how to work.
They should not choose
whether policy applies.**

OS-enforced sandbox ceilings:
a first concrete step toward governable agent action.

Thank you!



Agent Governance Lives in the OS