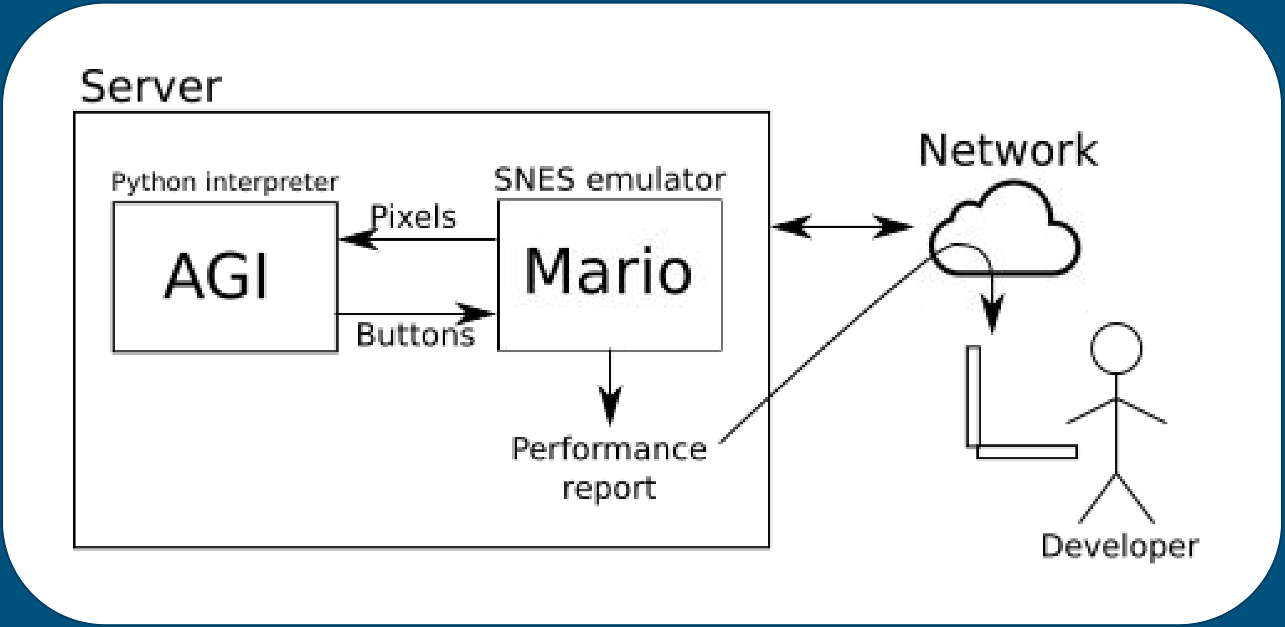
The AGI Containment Problem

So you have an untested, maybe-Superintelligent AGI. Now what?

Computer Security Applied to AGI

- Future AIs may be smart enough to discover novel security vulnerabilities
- Vulns aren't magic; we can identify their types and consequences and mitigate risk in advance
- For short-term improvement: fix vulnerabilities
  For long-term improvement: fix architecture

A Mario-Playing AI



A Mario-Playing AI

Convergent Instrumental Goals

AGIs with many likely goal systems will want to:

- Increase the probability that similar AGIs come to exist
- Take control of their execution environment
- Gain access to outside resources
- Score well on tests

All of which are potentially enabled by:

- Finding security vulnerabilities in their execution environment

What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- Accurate log files and debug information
- Reliable tripwires
- Reproducible execution
- Secure reset
- Test isolation

What is an AGI Container

A computing environment that enforces *invariants*:

- **It only runs the code you want, in the places you want**
- No unexpected input or output channels
- Accurate log files and debug information
- Reliable tripwires
- Reproducible execution
- Secure reset
- Test isolation

Avenues for Code Execution

- Intentionally-given permission
- Output formats
  - Eg CVE-2014-9495 (buffer overflow in libpng)
- Test environments
- Self-exploitation: AGI module 1 triggers a buffer overflow in module 2 by thinking a malformed thought

Terminal Emulation

Text isn't just text (if you view it in a shell); they respond to "escape sequences", a protocol for changing colors and moving the cursor around that evolved incrementally from the VT100 (1978).

> What can happen if you `cat` an AGI's output?

Terminal Emulation

CVE-2006-7236 xterm: execute arbitrary commands by setting the window title and then sending <CSI>21t which types the window title

CVE-2008-2383 xterm: execute arbitrary commands via LF characters surrounding a command name within a Device Control Request Status String (DECRQSS)

CVE-2015-5309 Buffer underflow in PuTTY via an ECH (erase characters) escape sequence

RowHammer

Making repeated memory accesses to two memory rows adjacent to a target row causes bit-flips in the target row

Y. Kim et al (2014) “Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors”

What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you think it’s running
- **No unexpected input or output channels**
- Accurate log files and debug information
- Reliable tripwires
- Reproducible execution
- Secure reset
- Test isolation

GSMem

Accesses memory in a pattern that creates radio-frequency interference, which can be picked up by a reprogrammed cell phone

M. Guri et al “GSMem: Data Exfiltration from Air-Gapped Computers over GSM Frequencies” (2015)

Cache Timing

Access to a clock (or a race condition) can reveal a surprising amount

- M. Weiß et al “A Cache Timing Attack on AES in Virtualization Environments”

What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- **Accurate log files and debug information**
- Reliable tripwires
- Reproducible execution
- Secure reset
- Test isolation

What is an AGI Container

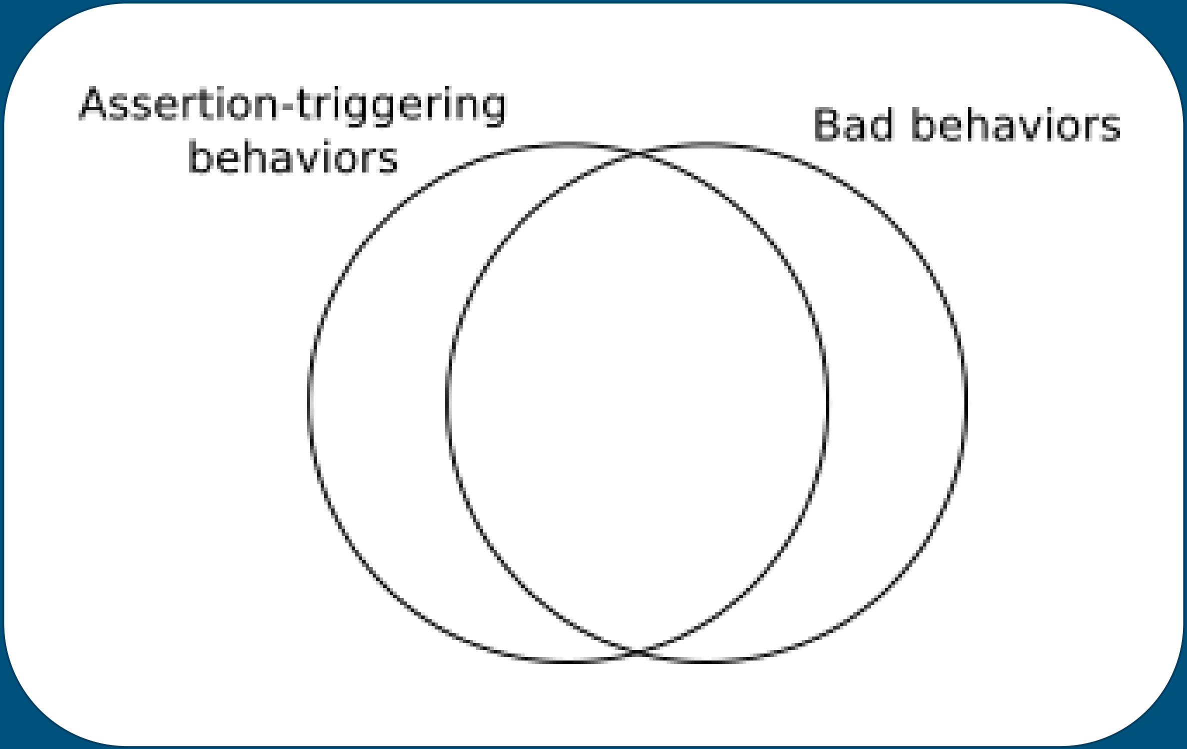
A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- Accurate log files and debug information
- **Reliable tripwires**
- Reproducible execution
- Secure reset
- Test isolation

Tripwires

- Semantic tripwires:
  - assert(“Not planning to take over the world”)
  - Depends on future algorithm details
- Technical tripwires
  - assert(“Data structures aren’t corrupt”)
  - Can be worked on now

Tripwires



What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- Accurate log files and debug information
- Reliable tripwires
- **Reproducible execution**
- Secure reset
- Test isolation

Reproducible Execution

- Devops: keep track of inputs, random seeds and build versions
- Algorithms: make stochastic algorithms less sensitive to the seed
- Language design: use languages that don’t have race conditions

What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- Accurate log files and debug information
- Reliable tripwires
- Reproducible execution
- **Secure reset**
- Test isolation

Hardware Persistence

Jellyfish: Malware that runs on a GPU; persists across warm reboots

BadUSB: Reprograms USB peripheral firmware to add malware back to a computer after it’s removed

IRATEMONK: Reprograms hard disk controllers to occasionally add code to programs as they’re read off the disk

What is an AGI Container

A computing environment that enforces *invariants*:

- It only runs the code you want, in the places you want
- No unexpected input or output channels
- Accurate log files and debug information
- Reliable tripwires
- Reproducible execution
- Secure reset
- **Test isolation**

The AI Box Experiment

Premise: One person role-plays as an AGI, the other role-plays as a gatekeeper

Successful “AGI”s: Eliezer Yudkowsky, Tuxedage

A Superintelligence should do just as well or better

The AI Box Experiment

Premise: One person role-plays as an AGI, the other role-plays as a gatekeeper

Successful “AGI”s: Eliezer Yudkowsky, Tuxedage

A Superintelligence should do just as well or better
...if it can have a conversation with a human

Information Hazards

Which AGI outputs are okay to be looked at, by whom?

- Depends on how persuasive the AGI might be (today's classifiers are no risk)
- Information that the AGI controls less directly is safer
- Outputs that the AGI optimized for a known (non-persuasion) objective is safer than output that it didn't
- Publishing is a much bigger risk than showing information to developers

A container's job is to enforce policy

Some Expensive Strategies

Information-theoretic containment

The strictest possible policy: Ask a small number of multiple-choice questions. If there’s only a few bits of output, there’s no room to hide messages or tricks

R. Yampolskiy “Leakproofing the Singularity”

Homomorphic encryption

Transform computations in such a way that intermediate steps are meaningless without a key (which is kept elsewhere)

Degrees of Containment

Light

- “We don’t have a working AGI yet, but we’ll use your containment software as long as it doesn’t get in the way”

Medium

- When you’re willing to pay large but reasonable costs to secure your AGI while testing

Heavy

- For when nothing but a provable guarantee of security will do

Mechanisms

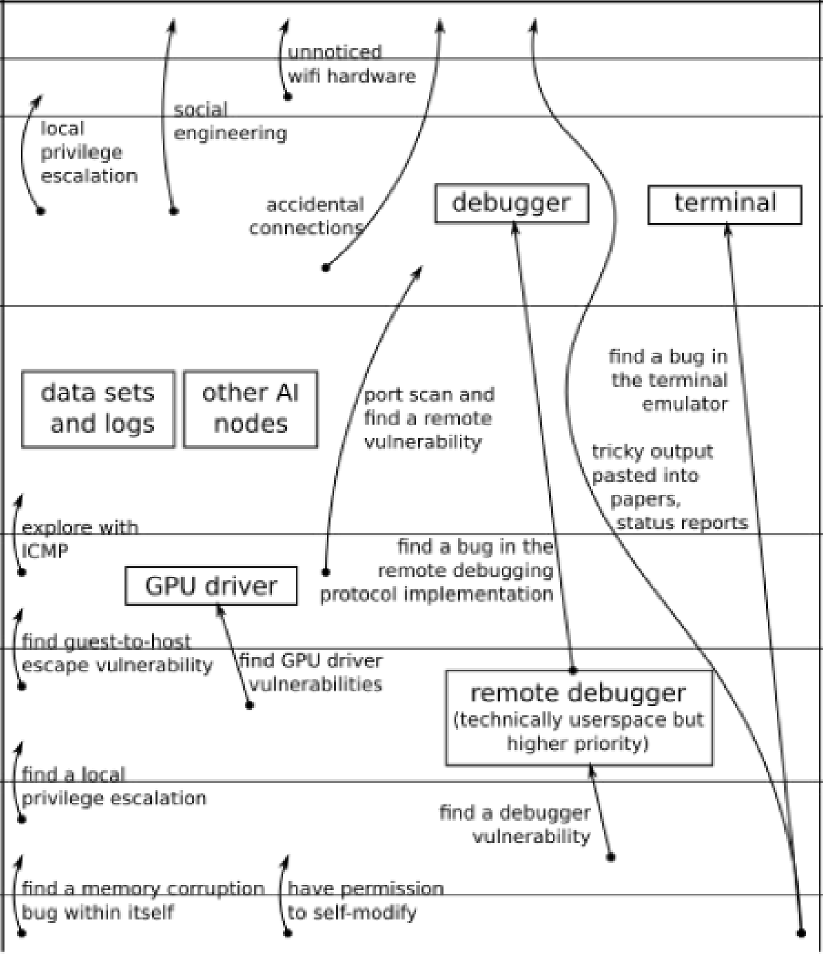
Language semantics

Operating system sandboxes

Virtual machines

Network topology

Layer	Components and Escape Strategies	Exploitation Consequences
internet-connected computer		• AI escapes onto the internet
airgapped workstation (kernel)		• AI sees all source code, notes about it
airgapped workstation (process sandboxes) The debugger, the terminal that views AI output, and everything else are separate process sandboxes.		• AI can monitor developers in detail
datacenter routing layer Realistic AGI will be distributed across many nodes in a datacenter for performance reasons. AI hardware is kept separate from general circulation and isolated from everything except researcher workstations by network topology.		• AI experiments no longer isolated from each other • AI gains extra information • increased risk of hardware persistence, RF attacks
VM host		• monitoring is severely impaired • soft reset no longer reliable • risk of corrupted hardware
VM guest (kernel)		• monitoring is impaired
sandboxed process		• AI can self-modify • log files generated by the AI program itself may be falsified
running AGI		



unnoticed wifi hardware
social engineering
local privilege escalation
accidental connections
debugger
terminal
data sets and logs
other AI nodes
port scan and find a remote vulnerability
find a bug in the terminal emulator
tricky output pasted into papers, status reports
explore with ICMP
GPU driver
find a bug in the remote debugging protocol implementation
find guest-to-host escape vulnerability
find GPU driver vulnerabilities
remote debugger (technically userspace but higher priority)
find a local privilege escalation
find a debugger vulnerability
find a memory corruption bug within itself
have permission to self-modify

Mechanisms

Language semantics

Operating system sandboxes

Virtual machines

Network topology

Language Semantics

A program that never calls a networking-related function can’t make network connections

- Unless it exploits a buffer overflow-style problem in itself or a library
- Or it exploits a hardware bug such as RowHammer
- Or it calls a code-execution function that wasn’t recognized as such

Mechanisms

Language semantics

Operating system sandboxes

Virtual machines

Network topology

Operating system sandboxes

Seccomp (basic version): A program declares “from this moment onwards, I will never open another file, or make any system call except reading, writing and closing the files I have already opened”

Seccomp (advanced version): Write a Berkeley Packet Filter program applied to Linux system calls for a process

Chroot: Restrict a program to a specific directory of the filesystem

Mechanisms

Language semantics

Operating system sandboxes

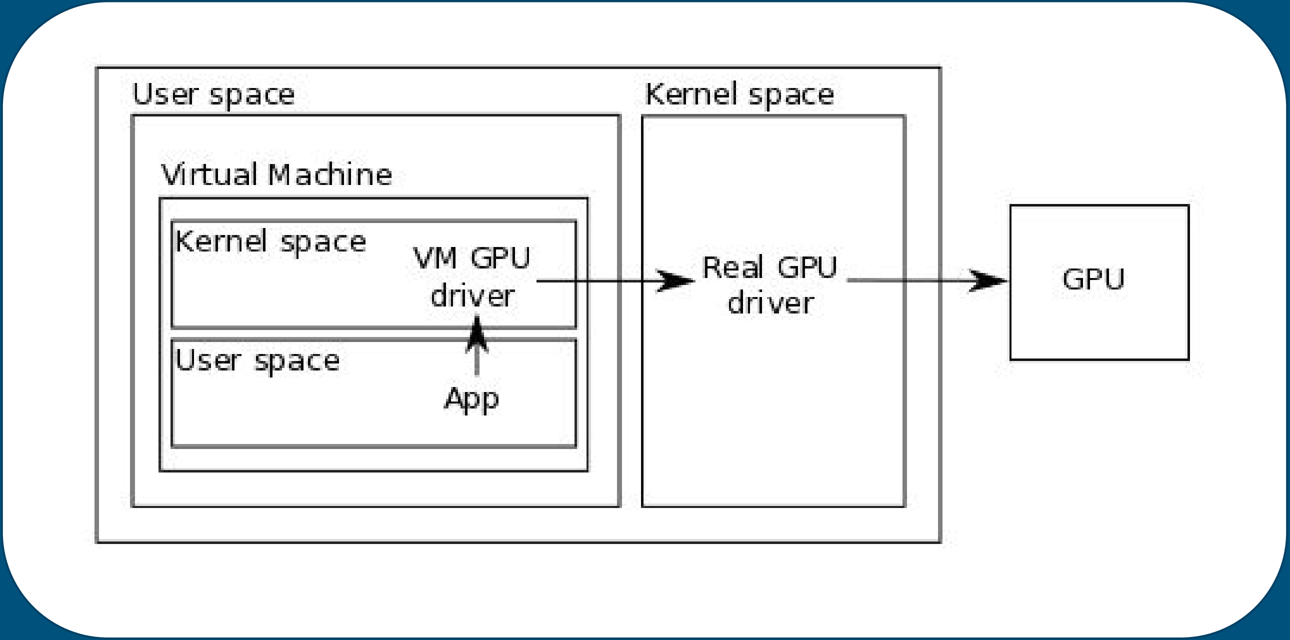
Virtual machines

Network topology

Virtual Machines

- VirtualBox, VMware
- Uses CPU features to create an OS-within-an-OS
- Security vulnerabilities found with fairly high frequency

Problem: GPUs



GPUs interact poorly with virtualization

Problem: GPUs

“The driver is gigantic. Think 1-2 million lines of code dealing with the hardware abstraction layers, plus another million per API supported. The backing function for Clear in D3D 9 was close to a thousand lines of just logic dealing with how exactly to respond to the command. It'd then call out to the correct function to actually modify the buffer in question. The level of complexity internally is enormous and winding, and even inside the driver code it can be tricky to work out how exactly you get to the fast-path behaviors. Additionally the APIs don't do a great job of matching the hardware, which means that even in the best cases the driver is covering up for a LOT of things you don't know about.”

- Promit (former nVidia driver developer, 2015)

Mechanisms

Language semantics

Operating system sandboxes

Virtual machines

Network topology

Network Topology

- If there’s no physical connection, there’s no attack surface (“airgapping”)
- In practice, expect all AGI experiments to be done on network-connected computers so developers can get work done
- Likely compromise: routers configured to limit where messages to AGI can come from and where messages from AGI can go to

What to do about it today?

- Code-integrity enforcement tools
- Automate configuring off-the-shelf software into boxes
- Harden libraries commonly used by ML
  - Stella, OpenAI Gym
  - Torch, Theano
- Red-team exercises
- Hardened debuggers