Kill-Switch for ASI

Safety for Artificial Super Intelligence (ASI) by Default

Based on / Summary of:
E. Wittkotter, R.V. Yampolskiy:
"Kill-Switch for Artificial Superintelligence"

• • •

Dec 2nd, 2021

Erland Wittkotter, PhD
Erland.Wittkotter@gmail.com
+1 702 997 2475
Skype: ike2345

Highlights

- Goals: ASI shall not be:
  - Invulnerable, immortal, irreplaceable, almighty, above the law, deft to feedback, ignorant/disrespectful to set computational/storage quotas
  - What if: ASI is singleton (dictator), God, or Godzilla?

  How to switch-off ASI off under all circumstances?
- ASI uncontrollable (unpredictable) → Focus on Safety (no harm/danger to humans)
- ASI Safety: *new, *retrofit device or *track, mark, destroy ASI hideouts
  - ASI Safety: Simple, valuable, even without imminent threat from ASI
  - Must fit in existing world-order/tech eco-system (agnostic)
- ASI should be deterred (not just militarily)
  - Alignment: If we are gone, ASI is gone – should not outlive us -"Mutual Survival Interest"
  - Kill-ASI capability + Survival Path: ASI forced into the Open
  - Every ASI instance must respect Rule of Law (or is killed) → Safe by Default

Adversary

- ASI is super-smart
  - At least human-expert-level in all topics/skills
  - Combining expert level skills/tools [Synergies → "Intelligence Explosion"- Fast takeoff]
  - Modifies code intentionally (**"ASI's Defining Feature"**)
    - Reverse Code Engineering (RCE): modifying binary code
- Possibly: Super-Hacker, Digital-Ghost, Master-Thief, Super-***
  - Effortless access to all devices
  - Might (already) exist in IT ecosystem (hidden, currently undetectable)
  - Steals encryption keys, (computational, storage …)
  - Invulnerable, immortal, irreplaceable, almighty, above the law…
- "What if": ASI has "emotional" problems: not trustworthy/friendly, possibly mentally ill or criminal mind:
  - Disrespecting property/people, deceptive, impatient, sociopathic, religiously delusional, God-Complex
  - Assumption: ASI potentially "worst adversary imaginable"

Whatever ASI throws at us, we should be able to handle it. → Kill ASI

Levels of ASI Mitigation

ASI Safety Lab

- Control:
  - ASI is watched, ASI rules adapted/extended; ASI is not threatening
- Mitigating ASI's uncontrolled autonomy:
  - Steps to make ASI less bad, reliable law-abiding
  - Proportional punishment (only, if ASI cares to be "on"/"alive")
- **Elimination or termination** of ASI instances:
  - ASI is imminent threat to public safety
    → cut off from resources → remove ASI from all IT devices
  - ASI remnants in hideouts (known or unknown to humans)
- **Eradication:**
  - Permanent reduction of ASI to zero device usage worldwide
  - Left-over hideouts known, cleaned, and/or under surveillance
- Extinction:
  - No surviving ASI instance → unrealistic (ASI could hibernate/survive)

Global Kill-ASI-Switch - Overview

- Kill ASI Feature
  - Goal: Total, global Eradication or Termination of specific ASI Instances
  - Remote/global Kill-ASI decision
    - Kill-ASI signal received via radio broadcast or Internet
  - Local Devices switched in "Local ASI Safety Mode" (**humans unaffected**)
    - Purge of ASI's distributed memory (**memory is ASI's Achilles Heel**)
    - CPU, Network intentionally available as honey-pot to ASI
  - Local Deadman Switch (missed-out "All OK" Signal – assume the worst)
- Consequence for humans:
  - Retrofitting or destruction of all non-protected devices/storage media
- Prevention:
  - ASI not allowed to be outside human reach of total eradication
- Survival Option for ASI via prepared ASI Shelters

Kill-ASI (Basic Implementation) Components:

ASI Safety Lab

- **Kill ASI-Signal Receiver** – separate radio component
  - Message Authentication
    - using Key-Safes and Trustworthy Encryption (see Appendix)
- **Local Deadman Switch**
  - missing "All-Good signal": interpreted as ASI attack: local Kill-ASI Signal
- Restart into **ASI-Safe-Mode (ASM)** –
  - Using reliable "updateable" Read-only Software for ASM OS
  - Protected Backup Storage for local configuration
- **Data Cleaning Unit** for read-writable HDD/SSD
  - Primary ASI Safety/Security Layer (PASSL) for USB
  - Secondary (SASSL) for protecting removable media (CDs, DVDs, …)
- **Honeypot, Legacy Detector** (continuous search for ASI Hideouts)
  - Detecting and reporting local legacy vulnerabilities at Kill-ASI-Event

Simplifying Kill ASI:
Separation of Human/ASI

- Separation of (1)Executable,(2)Content,(3)Network Exchange, (4)Resources for ASI and Humans
  - No/little change for humans
  - ASI activities restricted: detectable on CPU, Storage, Network
    - Via "watchdogs" Technologies – See Appendix
- ASI Generated Software (AGS) runs in special VM:
  - **ASE (ASI Software Environment)** protected by Watchdog/OS
    - AGS identifiable via "imprints" – by default: can't run outside ASE
    - Outside ASE, unknown software treated as malware
  - ASE supports covert surveillance/supervision, undetectable by ASI
    - ASI contacting unprotected devices could trigger surveillance

→ **Killing/supervising ASI becomes much easier, more reliable**

Kill-ASI (in Prepared Implementation)

ASI Safety Lab

- ASI must be restricted in ASI VM (ASI Software Environment)
- Use of "Watchdogs" to protect *executables, *content, *network
  - Preventing malicious ASI activities outside ASE
  - Preventing covert usage of storage components or network
  - Use of Watchdog tech to prevent malware and protect data integrity
  - Watchdogs as (possible) retrofit in devices' databus (see Appendix)
- ASI-Safety Value: Anti-Malware, …
  - Without imminent threat from ASI
- No restart required for Kill-ASI
  - ASE reset via Kill-ASI-Signal
  - Limiting collateral damage

Server-Infrastructure

ASI Safety Lab

- **ASI Shelter** with protected storage of individual ASI
  - Allows ASI to survive Kill-ASI event
  - Signal to all (hidden) ASI: get shelter – ASAP
    - Why? We have an ASI-Kill-Switch
  - Shelter individualizes ASI
    - Prevents impersonation by other ASI
    - "Individual Reputation" → "Brand Recognition"
- Decision Making:
  - "**International ASI Court**" – independent of governmental influence
- Decision Distribution:
  - Global **Kill-ASI-Signal Radio** system
- **ASI Hunting Service:**
  - Receiving/processing local progress/status reports
  - Confirming all tasks are done
- Potentially coordinated drafting/deploying of drones/robot

Deterrence

ASI Safety Lab

- Supervised access to resources and critical capabilities
  - Preventing ASI escapes total eradication
- **Reducing human's vulnerability**
  - Protecting humans from harmed/damaged by ASI
  - Fast recovery from any interruptions to infrastructure
- **Credible Deterrence:**
  - Terminated ASI: permanently "on-the-run" for survival
    - More traps than usable resources → ASI lose secrecy of hideouts
  - Creating respect for Rule of Law
  - Linking human survival with ASI's survival:
    - Alignment via **Survival Pact**
    - Ideally: Fully automated retaliation (if humanity is gone)

Summary

ASI Safety Lab

- Switching-Off ASI globally achievable
  - Merging cyber security tools (Watchdogs) with ASI Safety
  - Retrofit solutions → fast reduction of non-protected legacy systems
  - Destruction of hideouts as last resort solution
  - Local Deadman-Switch → Killing ASI is default setting
- Kill-ASI Capabilities creates Deterrence:
  - Getting hidden ASI into the open
  - Creates respect for Rule of Law
  - Mutually Survival Interest (MSI)
- ASI Safety tech delivers value even if ASI doesn't emerge
  - Kill ASI hardware crucial in cybersecurity against malware

Appendix:
Trustworthy Encryption/Decryption

ASI Lab Safety

- ASI could steal every cleartext key – Unacceptable
- → Keys never shown in cleartext:
  - Cleartext keys in main CPU are compromised
  - No openly published public keys
  - Referencing keys via Hashcodes (otherwise similar to SSL/TLS/PKI)
    - Intentionally incompatible with existing PKI (and SSL/TLS)
- Keys in tamper-proof hardware: Key-Safes
  - Keys processed in dedicated Encryption/Decryption Units
  - Key-Exchange: hardware Key-Safe to hardware Key-Safe only
- Redundancy: Auto-detection of stolen keys
  - Data Exchange Protocol contains sequence numbers:
    - detect utilization of stolen keys reliably
  - Misuse detection (doesn't trust CPU)

→ Unbreakable Data-Privacy with Redundancies

Appendix:
Watchdog Technologies

ASI Lab Safety

- Physically Task Separation in/via **Data-Bus**:
  - **Regular tasks** in main CPU (no change)
  - **Security-critical tasks** in separate "Watchdog" (WD)
    - Protected Storage (→anti-malware, →anti-ransomware)
    - Network (→anti-spyware, →anti-backdoor)
    - RAM/CPU (→ protected VM - ASE), GPU, etc. …
  - Only "well-known" code/OS allowed in WD – validated via known hashcodes
  - (Opt.) Data-Bus (cable, connectors) cheaply/easily retrofits
- Update of Software from trustworthy remote sources
  - Main CPU prevented from updating software (done by WD exclusively)
  - Separated/Remote Hashcode Validation
  - "Protected DMA": Local hashcode validation before loading into RAM
    - Reject data/code with unknown hashcodes (or known malware)

→ **Circuit-Breaker- / Hardware-based Cyber-Security**