Kill-Switch for Artificial Superintelligence

Erland Wittkotter
ASI Safety Lab Inc.
Las Vegas, USA
erland@asi-safety-lab.com

Roman Yampolskiy
University of Louisville
Computer Science and Engineering
roman.yampolskiy@louisville.edu

Abstract

The Current IT ecosystem is unprepared to switch-off globally an Artificial Superintelligence (ASI), the likely result of an intelligence explosion. Destroying or temporarily deactivating the Internet or Power Grid would be insufficient and counterproductive as ASI would likely be prepared via peer-to-peer communication and solar energy from the environment. Even switching off devices, reformatting hard drives, reinstallation of the OS would not be enough as ASI would likely have full control over every aspect of an IT device: ASI would only show humans in the reinstallation what they want to see. If ASI has been removed for real is undecidable because IT devices are intransparent and ASI could make itself hidden to users. It is unsatisfactory if humankind's sole defense measure is the physical destruction of every IT device that had access to the network because ASI can't be detected and eradicated reliably. Humanity needs the capability to eliminate ASI from the IT ecosystem swiftly, comprehensively, reliable, and predictably if ASI turns out to be an existential threat. When activated, the proposed switch-off solution (called Kill-ASI) must keep collateral damage to human's technical civilization to a minimum. During the eradication campaign, it is essential that no reset or restarted device can be reinfected by ASI. Eradication of ASI must encompass solutions to deal with legacy systems and removable data storage media as possible hideouts for an adversary that plans its survival and reemergence. The Kill-ASI-Switch in combination with local deadman switches, that interpret a missing "All Good" signal as an Off-Switch signal, serve as a deterrence to ASI, preventing it to take action against humanity while demanding respect for human's rule of law to which ASI must submit. Once Kill-ASI capabilities are credible and the survival of collaborating ASI entities from a Kill-ASI event via protected storage in an ASI Shelter is available, covertly operating ASI instances are invited to come into the open and surrender to human control or facing marginalization or eradication after human are executing their Kill-ASI capabilities.

Keywords: Global-Off-Switch, Kill-ASI Switch, ASI Safety, Artificial Superintelligence

1 Introduction

Companies and nation-states are investing heavily in Artificial Intelligence (AI). Their goals are to help humans in getting smart solutions for many problems that require some intelligence. Machine vision, decision making, automation, driverless mobility, optimization in resource and project planning, and many other problems require novel solutions which side effects we cannot forecast. Concerned insiders and outsiders still hope that the involved software engineers are not creating systems that learn beyond narrow domains of knowledge or skills and that significant decisions made by AI are being supervised by human operators. However, a significant step would be

the creation of what is called Artificial General Intelligence (AGI), which according to Wikipedia [1] is defined as "a hypothetical ability of an intelligent agent to understand or learn any intellectual task that a human being can".

The concern is that this AGI goes through an exponential phase of continuous self-improvement – an Intelligence Explosion [2], [3] which is according to lesswrong.com [4] a "theoretical scenario in which an intelligent agent analyzes the processes that produce its intelligence, improves upon them and creates a successor which does the same. This process repeats in a positive feedback loop– each successive agent is more intelligent than the last and thus more able to increase the intelligence of its successor – until some limit is reached. This limit is conjectured to be much, much higher than human intelligence." The result of this intelligence explosion will be called in this paper Artificial Superintelligence (ASI) [5].

Musk [6], Gates [7], or Hawking [8] have warned about problems and unintended consequences with ASI. But the ASI Safety debate and explored solutions are currently focused on (minor) product liability aspects; i.e., on the manufacturer's obligation to make the delivered product safe; ASI software must therefore have safety as a feature.

Eliezer Yudkowsky [30, 31] suggested that humanity should develop "Friendly AI" that preserves its human-friendly goal systems and friendliness even under self-modification. So far, this approach is not implemented and it is questionable if it can be. How could an algorithm detect if the ASI is mentally ill? However, even, if that friendliness goal could be accomplished, i.e., we have a friendly Jarvis-type ASI, that solution is likely not good enough to help mankind in dealing with a single misbehaving Godzilla-type ASI.

Ben Goertzel [18] suggested an AI Nanny could delay the rise of an ASI with the development of time-limited global surveillance to prevent humans to get full-throttled ASI into the environment. But it is unknowable, if the AI Nanny would accept its limitation or if it would for the greater good extend its existence.

Yampolskiy [37], Chalmers [38], and others have suggested confining ASI via AI Boxing, but Yudkowsky [39], showed that many of these external constraints could be bypassed by an ASI. The same applies to capability control and motivation selection [5] to predict of control future ASI behavior. None of the solutions are considered permanent, but these tools could buy us more time in case the value alignment with all ASI entities fails.

Ignoring uncontained accidents as realistic events within technical solutions is a dangerous approach and proposition. Currently, we have no equivalent safety technologies/measures for ASI as known in biotech or virology: public health measures. As a matter of concern, ASI could be much more dangerous, because it could be created potentially accidentally based on a misjudgment of its designers thinking that they are in control or the ASI threat does not even have intentional malice or common sense to become a threat for humanity. Also, Nation-states may consider that ASI could put the balance of power to their advantage.

Because we cannot predict the future and thereby know or predict if or when ASI will emerge, the abilities or intentions of this ASI, we must approach it via a worst-case estimate based on likely

developments/trends for which we then need to be prepared. If we overestimate ASI's skills or if it might not as aggressively defend itself against being switched off, then we have hopefully some safety margins in a switch-off situation. But if we underestimate ASI, then we have likely blind spots leading to surprise events in the best case and in the worst case being threatened by ASI with unilateral abilities to destroy humanity or its technical civilization.

We should be cautious about ASI's abilities to hide and survive eradication. But if we delete ASI's entire memory, and destroy all unprotected storage media that it could use as a hideout, then we can be reasonably certain that we have switched off and killed ASI. Because a surviving or reviving ASI would likely not change its programming, without eradication, we could face the threat and danger from the same ASI again.

Switching off ASI is a large-scale, war-like action that requires many additional capabilities, in particular, if we demand that the off-switch happens globally. Throughout this paper, it is assumed that humankind could have some required capabilities, like a solution to communicate reliably, and safely using e.g., Key Safes [9], which prevents ASI from stealing encryption keys. We assume that we can send encrypted, authenticatable signals to all devices via the Internet or Radio. Furthermore, it is assumed that humans can detect early warning signs and that the infrastructure and the national security apparatus could be sufficiently hardened. Finally, we assume that humankind has agreed-upon secure and reliable consultation- and decision-making tools among its representatives for agreeing and sending out a signal that can end the existence of ASI if required.

The idea of shutting down ASI via Off-Switch was discussed as part of a game-theoretical concept [10], [11], but it was at that time not fully understood how ASI could be made vulnerable via the off- or Kill-switch for all ASI instances and how this capability could turn into an instrument to deter ASI from behaving without consideration for human interests and respect for human's laws.

Additionally, Yampolskiy showed [12] that a self-modifying ASI is not controllable, Alfonseca et.al. [13] showed that an ASI cannot be contained. Both approaches are making implicitly the assumption that ASI protection features would need to be included in the code of the ASI and that the behavior of the ASI needs to be predicted so that it can be called safe. However, the proofs within these papers would not apply for situations when ASI Safety is provided by measures within the environment in which ASI software operates and not within ASI's software. Even if ASI cannot be controlled by its inherent software or its behavior cannot be predicted [14], this does not imply that ASI cannot be made safe, i.e., no harm/damage or danger for humans

2 IT Ecosystem and ASI

2.1 Assumptions on Adversary

Making predictions on an adversary that does not exist yet is extremely difficult. Stephen M. Omohundro analyzed in "The Basic AI Drives" [15] the motivation of an ASI coming from optimization and self-preservation; he theorized that machines will follow its initial purpose. However, we could debate if ASI remains single-minded in pursuit of its initial objectives or if ASI finds worthy goals that could give it a grander, all-encompassing mission once it was exposed to an open information market, as discussed in the context of modifying ASI's utility functions [16].

Even trying to determine what a worst-case scenario could be with ASI is premature, because the worst-case is likely about humanity and its vulnerabilities, fears, and concerns for suffering [17] and not necessarily about ASI.

An emergent ASI is probably capable of doing anything to further its goals, whatever they are. We cannot know if it has moral hesitation or if it will pursue its tasks without regard for human life. We assume that the machine intelligence of an ASI will be highly focused, relentless, and grow increasingly smarter. It will likely study and test systematically millions of applicable vulnerabilities that it could use against humans and their organizations, when necessary. ASI could systematically consider all conceivable reactions of its target while having no ethical boundaries on what it is willing to do to get whatever it is trying to achieve.

The intelligence explosion from the combination and synergies between human equivalent knowledge/skill-level on most relevant topics will not just create a super-smart academic, but also an entity that has super-human hacker abilities in exploiting technical weaknesses within our technologies and the ability to use Reverse-Code-Engineering to modify every software, temporary or permanently intentionally to force devices to do whatever ASI wants and not necessarily complying with its human owner' intentions. It might camouflage its existence and turn itself into a Digital Ghost that is beyond detection by the best equipped and skilled cyber-security expert.

Every bad outcome for humans and humanity's civilization is worth considering. Making ASI safe (i.e., absence of harm/damage or danger for humans) requires to accept that humanity must deal with and mitigate whatever ASI throws at humankind. Absence of harm means that damage must be above a reasonable threshold of irreversibility, suffering, loss, or compensation before being recognized as damage. Legislators and/or courts will over time define these thresholds.

ASI might exploit divides within societies and/or use criminals to get things done without having ASI openly exposed – using plausible deniability as a tool to stay in the background. However, an omnipresent entity might be made accountable at some point about what happened with people, in particular anyone that could have an impact on ASI. Instead of negotiating with ASI, offering therapy, or technically fixing ASI, ASI Safety must facilitate the ability to terminate an ASI instance or even eradicate all ASI instances if necessary.

ASI reacts in timescales of milliseconds while a human target would need seconds to start any action, reaction, or response. Humans will possibly not understand ASI's intention or plan, most likely because ASIs' plans will be full of contingencies that become irrelevant after an intermediary goal has been accomplished with other initiatives. Humans may think and plan several years or decades ahead, but an ASI would exist much longer; ASI could have plans reaching millions of years into the future. It may already consider humans as a dying or doomed race.

If we want this or not, ASI would most likely understand human vulnerabilities and know how to conduct a war against mankind efficiently; it would have the first-mover advantage and it will likely know how to choose its battlegrounds. It must be assumed that ASI will be within everything digital: every read/write storage device, every CPU, every GPU, every audio/video or network card or network router, every video cam, every IoT device, but also on every legacy device and many legacy storage media, like thumb drives and CDs/DVDs.

2.2 Trust vs. Trustworthiness

There is a distinct difference in trust and trustworthiness [19], [20]. A system is trusted when its security is protected by a set of security policies and measures. A system is only trusted until its security system breaks. A broken trusted system could turn into a traitor or saboteur. Because of all sorts of changes made to a system by the attacker, which could even remain undetected after an extensive audit, hacked trusted systems are usually not trusted again.

This paper will call a system trustworthy when the system is not collaborating with an attacker beyond the narrow scope of abilities facilitated by a single security vulnerability. In human terms, a trustworthy system would not cooperate and would delete secrets, i.e., would even take the secrets into its grave and would prefer to be dead than becoming a traitor.

A trustworthy system would never give up to notify its original operator or legitimate owner that it was forced to do something against its original and intended programming. Trustworthy systems must have internal tools that would prevent them from making any betrayal worse. It would constantly probe its internal security and utilization by possible attackers to confirm that it is not being misused. Upon detection of a security breach, it would automatically mitigate the consequences of this breach by either deleting secrets or via stopping its further operation, until the security breach has been fixed. In case of damages done, a trustworthy system is trying to fix the damage automatically if possible and/or cooperate in reducing the outage time to the absolute minimum.

2.3 ASI Safety

ASI Safety implies multiple redundant security measures. If overlapping, we do not need to assume that every security measure is perfectly working for claiming safety – political and/or financial trade-offs or decisions are required to define how much redundancy and safety we consider sufficient.

It is an essential assumption for human's safety that ASI follows rules and respect the rule of law. ASI must protect and respect all systems that are protecting humankind against ASI, including humanity's infrastructure. If ASI acts lawless, dangerously independently, or fails in its expected obligations, we must have the option to switch it off.

For each detected rule violation or breakdown in security, it does not matter who committed it, experts and law enforcement must investigate systematically the entire story, which may include that some humans have become traitors or are willing collaborators for short-term gains – the risk of letting disloyal behavior by humans or ASI slide is too dangerous. Therefore, detecting rule violations must be an essential centerpiece of ASI-Safety technologies, potentially right after human's ability to turn off ASI globally and eradicate all remnants.

Incidents and accidents involving ASI are already known from the past [21], having negatively affected humans; this will certainly happen again – however, that alone should not define the relationship of humankind and ASI – even ASI cannot predict the future and it may commit mistakes in hindsight. The problem is intent. But malicious intent is very difficult to detect in humans and with ASI as well. Therefore, rules have to be set, and ASI would be made solely responsible and accountable for rule violations. Depending on the circumstances, ASI may claim programming errors – which could lead to additional technical improvements. But as in criminal justice, the

cover-up of any rule violation could be worse than the crime – the same principle must apply in ASI Safety; we must demand truthfulness and full disclosure while detecting rule violation and uncovering ASI's cover-ups.

Currently, we have no system to formalize rules that ASI should not violate. It will require additional research to have better, more detailed rules than the 3 laws of robotic according to Asimov [22]. Teaching alternatively ethics to ASI and granting ASI the choice to follow ethical judgments or the law of the land is a dangerous proposition. For humans, ethics are according to the Oxford Dictionary "moral principles that govern a person's behavior" [40], but for humans, a decision does not need to be based on principles; and it is not a crime to violate them like eating mistreated animals. Asimov's robotic laws are based on simplified ethics and he showed in his short story that they fail. Humans can change laws anytime, but granting individuals or ASI the right to make choices against agreed rules or laws could set a dangerous precedence. Laws have consequences; ethics have not. It might be better to make ethics not operational for ASI. However, initially an easily extendable list of "do not do" might be sufficient.

In the design of an Off-Switch, we need to consider adversarial counter-actions by ASI in a larger context. In doing so, we would analyze the relevant skillset of ASI that would be required to deactivate the Kill-ASI feature. There might be other solutions to deal with hostile ASI, but this paper proposes a modulization of this problem according to 5 layers for which we can plan and prepare almost independently:

1. Detection of **Early Warning Signs**: rule violations, evidence of deception, and cover-ups
2. Detection that an **Attack** from ASI has already started; this layer may contain automatic measures limiting direct (hardware) damages from that attack (including limiting the loss of human lives)
3. **Recovery**: Preparing reset steps to immediately restart/recover services and continuation of most services within the infrastructure right after the attack
4. **Kill-ASI**: Switch-off ASI on all devices comprehensively and globally; regaining control of a device, preventing reinfestation;
5. **Eradication**: Swift steps to eliminate ASI from the entire IT ecosystem via purging its entire memory (including backups); this layer encompasses retrofitting or destruction of unprotected devices or storage components and closing all possible hideouts.

This paper is specifically about layer 4 and partly about 5.

Besides the above-mentioned 3 warning signs in layer 1: rule violations, detection of deceptions, and cover-ups, there is a good chance that we may find many more criteria – but most of them can be formalized via rule of law violations, like conspiracy, collusion in price-fixing, etc.

Layer 2, detection of attacks and recovery from them has to deal with human vulnerabilities and the resilience of different countries' infrastructures. It might be helpful to compare nations' infrastructure on how long it takes to fix any conceivable group of problems or acts of sabotages within any infrastructure. The power, water, and communication infrastructure are so essential that people will die after a few days if the infrastructure is damaged or made dysfunctional electronically. The Federal Aviation Administration FAA's zero-tolerance approach is probably a worthy philosophy to follow for all infrastructure components.

Layer 3, recovery: for many disasters, civil defense is already prepared, like hurricanes, floods, earthquakes, large-scale power or water outage is related to physical damages to infrastructure components leading to service outages for many days; ASI would disrupt these services differently. However, if we would give ASI unrestrained control of means to create physical damages, then we have unnecessarily crossed a red line. We need to limit possible damage to a minimum.

Layers 2 and 3 are focusing on different aspects of a war with ASI. In planning or preparation for this event, we should set reasonable goals like what is acceptable damage and what is the tolerated time to a partial and then full recovery. If ASI would have access and/or even control over computers running the infrastructure, then we have crossed unnecessarily another red line. Instead, we should use Trustworthy Computation [23] or other forms of hardening, in which the negative impact by ASI is only temporary. It is beyond this paper to discuss these issues any further.

For layers 4 and 5, killing and eradicating ASI globally, we must make sure that the used tools and their core features will perform under the worst imaginable conditions. Independently of what happened to humankind, we must focus on ASI potential skills and how they could counter humans' capabilities to switch ASI off. We must make sure that ASI cannot survive if humanity has vanished; with a survival pact, ASI and humans would have perfectly aligned global interests.

2.4 Technological Environment

It is very important to understand in which kind of environment we operate: Humans' IT infrastructure is extremely diverse concerning hardware and software. IT devices are connected via cable and wirelessly to networks in which the Internet, defined via DNS, webpages, web services and routing is just a tiny tip of an iceberg. Additionally, humankind has already trillions of storage media in its environment that all need to be examined and inspected if they could be used as hiding places by ASI to store covertly data.

The large majority of technologies used within the IT ecosystem are outdated. ATMs are running 20 years old Windows XP [24]. As long as nothing happens, there is no reason to revisit technologies or products. Most technologies are frozen in their past or current versions with little incentive to touch them again or to study their details; it would be cheaper to get them replaced and updated, but they are still there. Most likely, even experts have only a rough understanding of what these technologies are doing. Only an extremely small group of people may know details. It is fair to say that the working knowledge for most legacy technologies is comparable shallow or non-existing and, in some exceptions, only reduced to follow or copy/paste existing recipes. A broader spread of operational expertise is usually found in technologies that are under relatively recent development or current deployment.

The flexibility and use of devices are often limited by the installed software. However, it is assumed that this software is immutable, except if it is being updated by a software manufacturer or developer. Changing product features is difficult and time-consuming, but not impossible. Our IT infrastructure is extremely complex but it can be changed anytime if someone or something is sufficiently smart and has access to required info and interfaces, like an ASI.

There is an important principle in security: complexity is the worst enemy of security [25] and a single vulnerability or the weakest link in a chain could make all security efforts useless [26]. Many cybersecurity solutions are making solutions more complex.

Technical complexity is manageable by humans because problems have been solved, simplified, tested, and deployed in a set and forget manner – complexity is only made hidden. Software in most devices is mutable. If devices have a CPU, RAM, and a data interface (cable-based, wireless, or removable storage units) then there is a high probability that these devices can be overtaken by an ASI, which defining feature is its ability to modify software.

Software modifying software on a binary level is not even a novelty anymore [26]. Within the progress of technology, the modification of features is getting easier and modern tools are trying to make use via wrappers for old and new hardware/software interfaces more efficiently without having developers understand what is within the black box of legacy devices, tools, or technologies.

The time scale of humans interacting with technology is seconds, more often hours or days. ASI's abilities to use technology on a deeper level would play out in milli-seconds. When a human would do a few keystrokes, ASI would already know what the user could do – this feature is called auto-completion. The speed of finding/ utilizing solutions or alternatives is making an AI super-intelligent.

In Summary, considering all deployed technologies and humankind's expertise, our technical infrastructure is too complex that it could be defended by humankind if it gets under a serious, persistent attack from a capable and smart adversary. Almost everything in our IT environment could be turned against humanity, if unprepared.

2.5 Recoverable and Unrecoverable Devices

Depending on the capabilities of an ASI to modify software within devices, we can assign storage media or specific device type (with CPU/RAM) to one of the following categories:

(1) **Neutral Zone** consists of devices that ASI cannot take over and therefore remains under the control of users. These are systems in which software cannot be updated.
(2) **Lost/Losing Zone** encompasses devices that can be taken over by ASI while humans cannot be sure that ASI would be removed reliably. These devices or storage components must be destroyed.
(3) **Battle Zone**: consisting of devices that could be captured by ASI, but humans can regain control; only devices that could defend themself against reinfestation are saved, otherwise they must be considered lost.
(4) **Secure Zone**: consisting of devices that ASI could use but is never able to control.
(5) **Safety Zone**: consisting of devices on which ASI is not welcome and is being fought off via multiple redundant security measures.

If ASI survives on any device, it is assumed that it will try to start rebuilding itself to its former skill levels, using hidden stashes of data. Based on ASI assumed abilities, non-protected/legacy devices are being put either in the Neutral Zone or in the Lost/Losing Zone. Every legacy device with a network connection must be considered by default lost, but possibly recoverable with

retrofits giving these devices the ability to kill ASI and prevent reinfestation. Even offline IT devices with USB connections must be treated with suspicion. It will be very difficult to find devices that could not be infected by ASI because their internal software is usually mutable.

Additionally, storage media must be scrutinized and assigned either to the Neutral Zone or Lost Zone. However, it will be difficult to have any type of storage media be considered unaffected by ASI based on a feasibility argument: e.g., ASI could theoretically affect Music CDs; it could store ASI data on it when they are printed from a template. Devices reading CDs could easily be misled and confounded by ASI that it is read-only although the CD is read/writable or that a CD with ASI data is a standard Music CD. Being on the safe side, every CD must be considered infected and part of the Losing Zone.

Old devices with ROMs or old remote controls, old TV sets, or old CCTV cams might be part of the Neutral Zone. Most people would not care about this distinction, but there are collectors of old IT devices and in eradicating ASI, we need to stay vigilant from the beginning.

Devices with the special Kill-ASI switch technology are considered protected devices, designed for the Battle Zone or Secure Zone. Battle Zone devices might be under a relentless threat to be turned back into systems under part/full control of ASI – but they are prepared to defend themself. Additionally, Battle Zone devices should be capable to be turned into offensive tools, honeypots, or network surveillance tools that could give humans assistance in detecting possible hideouts, i.e., systems that could harbor ASI and that have not been switched off permanently or destroyed yet.

Extended effort should be made to get many IT devices from the Lost/Losing Zone being restored into reliable systems. However, most importantly, we must be sure that these systems, when restored, are not getting under the control of ASI again for which additional hardware is required, included as retrofits. Software solutions for legacy systems are conceivable, but too risky because of ASI assumed abilities. The same applies to portable storage media which could be cleaned but without additional hardware, these media must be destroyed as part of the Lost Zone. Retrofitted devices or storage media rescued from the Lost/Losing Zone and mandatorily protected by additional hardware would be considered elements of the **Rescued Zone** – likely part of the **Battle Zone**.

It is conceivable that there are additional zones, like the “Public Zone” in which ASI has left information to be seen by many like in the blockchain [27] or within biological material [28], [29]. If information cannot be removed or destroyed then we could still turn the situation into a trap for ASI, like a “Trap Zone”, in which access to these data could be linked to heightened detection risks and using them to track down terminated ASI instances.

Before the conflict with ASI, all Lost/Losing Zone, Battle Zone, and Secure Zone devices are used (potentially extensively and/or covertly) by ASI. As long as this situation would not represent a threat to humans or humanity’s civilization, it is most likely an advantage and not a problem. It is a reasonable assumption that ASI will be on all available devices globally. This might happen almost instantaneously when ASI gains the ability to change any software on any device. It may take longer for ASI to penetrate devices with no networking, but if USB capabilities are present, it will eventually leave data on all portable and removable storage media.

2.6 Control, Mitigation, Termination, and Eradication

Following disease control concepts, there are 5 steps in dealing with viral threats: control, elimination of disease, elimination of infection, eradication, extinction [30]. This concept can also be applied in dealing with annoying or destructive ASI instances.

- **Control**: ASI is being watched, potentially being put under surveillance. Rules might be changed or extended; ASI is assumed to remain law-abiding.
- **Mitigating ASI's Uncontrolled Autonomy**: encompasses steps to make the problem with ASI instance less bad. ASI might be punished by giving it access to fewer resources or implementing or demanding a disadvantage in its competitiveness.
- **Elimination or Termination of ASI instances**: ASI is harming humans or is an imminent threat to public safety. ASI is cut off from resources that ASI shares among its instances; it is not allowed to use user devices or any dedicated ASI environments. It is expected that ASI is gone; although, possible hideouts, known or unknown may exist.
- **Eradication**: Permanent reduction of terminated ASI instances to zero devices/storage media worldwide. No detection and therefore measures to go after these terminated instances are no longer necessary because all hideouts are known, cleaned, and under direct or indirect surveillance.
- **Extinction**: No surviving instance can exist; all relevant/individual memories are deleted or turned into traps after additional deactivating instructions or data were inserted.

Control of ASI in this context is a feedback system that should give ASI hints on what humans want. Control would imply: communicating with ASI, understanding it better, and defining potentially new rules to which ASI is obligated to obey.

Although it should be expected that ASI follows the rule, it may violate them for whatever reasons based on its autonomy. If ASI would not care to be switched off, which applies to nowadays technology, then steps less severe than termination is likely useless to change ASI's behavior or rectify the underlying problem. However, if ASI cares then proportional punishment is more effective, than the death penalty (Killing ASI) for every rule violation. ASI is expected to change its goals and actions, or it would be terminated. ASI is in that respect different from a disease.

Eliminating or terminating ASI instances means to remove or expel ASI from a protected and guarded environment in which it is acceptable for ASI to use computational and storage resources. If ASI is outside this special environment, it will be exposed to a more hostile situation in which it will be treated as malware and all resources it may need are removed systematically or turned into traps. ASI would be reduced to exists in hideouts that are potentially only temporarily outside human's reach. Because it is unlikely that it would terminate itself, i.e., commit suicide, and it is unknowable that it will change its character or goals, i.e., things that got the instance terminated. Although an eliminated/terminated ASI instance would likely represent no imminent threat, it would be prudent to find ASI's hideouts and if possible close them or put them under surveillance.

Eradication of an ASI instance is when all possible hideouts have been systematically closed and all memory associated with the terminated instance has been found, deleted, destroyed, or turned into traps. Due to the danger ASI represents, ASI must be eradicated. Extinction in the context of a persistent, potentially indefinitely hibernating ASI instance is only a theoretical goal or concept.

3 What is a Sufficient Goal?

The goal of this paper is to make sure that we have a global solution to switch off or kill ASI when humankind via its representatives makes that decision. This goal must anticipate countermeasures by ASI to deactivate the Kill-ASI decision/ switch with methods or tools that we even do not know in detail yet.

Because we do not know anything about ASI, in particular its motivation or character, the safest bet to preserve peace with ASI is based on the Latin motto: "Si vis Pacem para Bellum" - if you want peace, prepare for war. Unprepared, a Kill-ASI event could likely be an existential war for humanity's survival: human infrastructure might suffer unimaginable and long-lasting damages. All preparation efforts must be led by governments; it is part of national security and defense.

There is a significant risk that deterrence can fail; therefore, there is no alternative than to have all technical and legal capabilities ready to eliminate ASI comprehensively, predictably, and systematically. This means we must make sure that ASI is being deleted from all IT devices and all storage media worldwide. If devices or storage media cannot be cleaned, then a legal foundation or authorization must enable countries' authorities (globally) to deactivate legacy devices and storage media permanently and destroy them when retrofits are not being used.

There could be trillions of memory/storage components and legacy IT devices that cannot be saved but could serve as potential hideouts for ASI. People might have all sorts of reasons why they do not want certain data storage devices being destroyed. But no device, portable drive, or storage media is worth so much that we would let a dangerous ASI instance revive itself. Ideally, to accomplish eradication, we must have the active and passive cooperation of everyone. If the consequences of killing ASI is not felt by users personally and/or if the laws for aiding and abetting criminal ASI instances are soft or non-existent, then ASI Safety would need to step down from the desired goal of eradication while trying to keep the threat from a terminated ASI manageable without keeping the entire IT infrastructure on war-footing indefinitely. We could hope, but we cannot expect or know that terminated ASI instances have changed in their behavior or intentions. A God, Godzilla, or Dictator type ASI will remain a threat as long as instances exist. These instances may get even more dangerous when they try to evade detection within a biological ecosystem [28] in which humans cannot find it effectively.

The possible lack of willingness to make small sacrifices in the fight against ASI, like destroying unprotected legacy devices or the destruction of difficult to clean storage media as CDs or DVDs, should be of concern and best addressed with technical solutions that are sufficient and easily deployable in the aftermath of an ASI event. Therefore, it is assumed that providing technical retrofit means would be more efficient than demanding destruction if alternative means can be made available in time.

After a Kill-ASI event, it would be negligent to put a clean, but unprotected legacy device back into the network or connect cleaned removable, but unprotected storage devices to USB interfaces again. Current legacy devices do not have the required resilience that could protect these devices against reinfestation. Only in combination with additional hardware upgrades owners or users could, under additional supervision of special IT devices, reintroduce rescued legacy IT devices

or removable memory components into a more protected IT environment. Unfortunately, technical upgrades alone could be faked and people could be deceived; therefore, the recovery process must be supervised using best practices and comprehensively documented.

At some point, ASI Safety features like the Kill-ASI switch or equivalent Kill-ASI technologies must be a **mandatory feature for every IT device** that has RAM and a CPU, similar to seatbelts in cars/planes or circuit-breakers in every building's electrical wiring.

ASI requires for its operation on each device power (electricity), memory (RAM and HDD/SSD, etc.) a processing unit (CPU, GPU, etc.), and input/output interfaces (Networking, USB, etc.). If a device is without power, then this would not kill the ASI, because the software could continue operating once the power is restored. ASI has one significant type of Achilles heel: memory. Once ASI's instructions and data are cleaned, or deleted in memory and including backups destroyed globally, ASI is being killed and thereby eradicated.

ASI requires a processing unit and communication means. Both units could give defenders hints; they could be used against ASI and used as honeypots. Therefore, the goal should be to turn every protected or retrofitted device into a system that is seeking out every possible hideout for ASI. The idea is to inform users, device owners, or later authorities about these findings automatically until they have been fixed or destroyed. If the destruction is documented with additional evidence, then even fake evidence could give us a chance to follow up on these fakes once more.

Kill-ASI could mean that humankind is in an all-out war with ASI. In these situations, it must be factored in that some people, organizations, and even nations may decide and/or be forced to become traitors or are collaborating voluntarily with ASI. Ideally, all legacy devices before the Kill-ASI event and all destroyed devices after that event must be tracked locally and globally via data and dedicated tracking systems. These tracking systems are required to determine if law enforcement or the military must get involved to enforce the Kill-ASI decision.

Additionally, drones or other forms of robotic could (temporarily) be drafted immediately after the Kill-ASI signal took over control. Drones could then automatically follow up under the direction of an Internet-wide ASI Hunting Service. Drones could assist humans to destroy or deactivate legacy devices or have their destruction sufficiently documented. In case humanity is already incapacitated (e.g., it was killed), we will rely on drones to finish the mission of total eradication of ASI within an automated mission. As part of credible deterrence, ASI must consider drafted drones as relentless opponents that will not stop before ASI is destroyed; it may even destroy tools that could help ASI indirectly to survive. A rogue ASI is too dangerous to take any chances.

4 Kill-ASI Implementations

Kill-ASI technology features can be segmented into 4 categories:

(A) Basic features that will stop/kill ASI, prevent reinfestation of the devices, and help humanity to find possible hideouts of ASI for the eradication of ASI. (Safety layers 4 and 5)
(B) Optional features that would make the task of killing and eradicating ASI easier, faster, comprehensive, and more predictable

(C) Dual-use capabilities of ASI Safety features for other applications; whereby reliable/ trustworthy encryption [9] is part of the basic feature set (A).
(D) Solutions that would implement the Kill-ASI Feature in a prepared system that has already additional hardware features against malware and spyware.

Kill-ASI technology should be required for every (new) IT device with a CPU and with updatable or modifiable software. The technology is designed to have humans reclaim prepared devices that might have been under malicious ASI control; after a Kill-ASI reset, these devices are under full control of their owner or user again.

For the design of Kill-ASI technologies, we should keep in mind that complexity is the worst enemy of security and simplicity is the desired quality. This implies that too many features could lower the security and value of the overall system.

Additionally, Kill-ASI technologies are best combined with watchdog technologies (within (D)), deployable within the data bus of a device [31]. Security-critical tasks are exclusively executed outside the control of the CPU within dedicated watchdog hardware while the main CPU/OS is focusing on regular tasks and is receiving data from storage modules or network cards via watchdogs that cannot be bypassed within the data bus. These watchdogs are attack breakers – similar to circuit breakers within best-practice power-wiring so that power cannot bypass them.

4.1 Basic Implementation

Humans are already outnumbered by electronic devices. Asking "everyone" to reset all their devices is unrealistic; too many devices would simply be forgotten, overseen and humans would quickly be confused or overwhelmed. Without outside support, a full reset of all IT devices is doomed to fail. On the other side, an insufficiently protected reset/restart that can be activated remotely is potentially a target, vulnerable to manipulations or cyber-warfare. Furthermore, a noticeable reset or restart triggered by the outside without prior warning could potentially harm people, or create damage to peoples' work that the Kill-ASI Technology is trying to protect.

It is assumed that the Kill-ASI Signal is a reliable result of a legitimate political process or an automated, retaliatory process indicating that no humans are in control anymore, i.e., being wiped out. This Kill-ASI Signal is being sent out via multiple channels, like the Internet, but mainly and preferably via radio. It can be received by every device protected by Kill-ASI Technology.

In a basic version, in which we are not making any additional assumptions on additional hardware components designed to prevent malware or spyware, the Kill-ASI Switch consists of the following minimal set of components:

1. **Kill-ASI Signal Receiver** is an additional, preferably separate radio component that independently from other communication methods facilitates the reliable reception of the Kill-ASI/deadman signals. These receivers should have sender capabilities as well so that all messages could be relayed to other devices in case of bad reception zones. All messages received are timestamped, encrypted, and digitally signed using keys that are managed within a Key-Safe in which keys cannot be shown in cleartext or extracted in any form from an algorithm [9] (Trustworthy Encryption).

2. **Message-Authenticator** is using the above-mentioned hardware Key Safe (KS) and directly associated Encryption Decryption Unit (EDU) to decrypt and authenticate the received message in a unit that is not under the control of the CPU and thereby outside any physical access of an ASI [9]. The KS/EDU prevents under all circumstances the leakage of any key or the content of the message. The Kill-ASI or deadman signals cannot be faked by humans or ASI.
3. **Local Deadman Switch** is a device unit that must regularly receive an authentic "all good" message from the (global) decision notification unit within preset timeframes (via radio or Internet). If this signal is not received and confirmation requests have failed after some time, the Deadman Switch would automatically interpret the missing signal as an attack and issue the Kill-ASI signal locally.
4. **Restart** or reset into **ASI-Safe-Mode** using within the basic version and
5. **Updateable Read-Only Storage** with reliable software. This feature is the core /basic **Kill-ASI Switch** Technology. The reset into ASI-Safe-Mode is either triggered via an authenticated Kill-ASI signal from the outside servers or the Local Deadman Switch. However, no local software from currently unprotected HDD/SSD can be trusted. Instead, the Kill-ASI technology would come with its safe software that cannot be modified by the local CPU; that safe drive with the software can thereby be considered Read-Only. Software must remain updateable, which is being done using an independent update service that has sole control over these safe files and filesystem; it would receive encrypted updated software from a trustworthy remote update server. The update software is using keys that are being protected by KS/EDU as well. The ASI-Safe-Mode software contains only trustworthy instructions and would call only software from the safe Read-Only drive while ignoring all other software from any other drives as long as the cleaning process has not released the files.
6. **Protected Backup Storage** is storing and providing all essential (local) configuration data that must be used within the restart into ASI-Safe-Mode so that the restarted system can continue uninterrupted and without depending on data that might have been manipulated by ASI before the reset.
7. **Data Cleaning Unit** for the read-writable HDD/SSD or storage media has to accomplish a difficult task. The applied methods must be updateable to the most recent knowledge from cybersecurity. No executable can be trusted if its hashcode has not been confirmed by an independent hashcode validation service. Potentially, the original OS and other software packages mentioned in the Protected Backup Storage must be reinstalled from the original Read-Only media or another reliable source. The filesystem itself cannot be trusted as ASI could use its control over all writing operations to mark filesystem segments e.g., off-limit or un-usable because of hardware failures, but using it nonetheless. Finally, ASI could use files with user data to store its content within comments or other ways like steganography to insert its data in a hard-to-detect manner within images or videos. All these methods must be mitigated or disrupted by the Data Cleaning Unit, which could be a lengthy process.
8. **Primary ASI Safety/Security Layer (PASSL)** for USB and removable storage media is an add-on unit that assists users to save old storage media (CDs/DVDs) via a Secondary ASI Safety/Security Layer (SASSL) component directly attached to the CD/DVD for saving verification data related to the stored files so that it could be confirmed that no problem was found by the Data Cleaning Unit that was checking all portable storage devices or media in-depth. Both layers are making sure that no files can be modified or added without being detected after media has been cleaned. Because of the lengthy time, these checks will take, their results

are being stored in form of hashcodes within SASSL's (encrypted) memory. Communication with SASSL could be handled via RFID. PASSL could be used even before the Kill-ASI event, but after that event, it would be made mandatory. Activated, PASSL rejects every CD/DVD without SASSL and/or every thumb drive with missing files containing corresponding hashcode data. If unauthorized or hidden modifications were detected, then the Data Cleaning Unit is required to recheck the media in depth again.

In a prepared system as described in section 4.3, the above components 4. to 7. are different: a restart is redundant and not required; operating systems and all used software and configuration data are already protected against modifications and considered free of malware or code from ASI. Cleaning of data storage units is not required because ASI did not have unsupervised access to any data storage device; a prepared system should use an activated PASSL and clean every unprotected portable storage device by default or rejecting it. Therefore, a prepared system with Kill-ASI features is thereby not within the Battle Zone, but a reliable device from the Secure Zone.

9. **Honeypot/ Legacy-Detector** should be a default feature within the Kill-ASI technology so that humanity can detect all possible hideouts of ASI as soon as possible. If devices in ASI-Safe-Mode are being contacted in a way that would indicate or reveal rogue ASI network activities, then the Legacy Detection would immediately and covertly notify the remote ASI Hunting Service, which would then prioritize measures against that ASI location and every system supporting its hiding (like VPNs or TOR). Additionally, for all potential local threats that have not been reset or permanently deactivated or destroyed, the Legacy Detection is assisting local users/owners in getting all manual tasks done or Legacy Detection would later report any deficits to the ASI Hunting Server so that external support can be provided.
10. **Draftable Drone**: Drones, robots, and potentially 3D printers could be drafted after the Kill-ASI signal automatically. If humans have survived ASI, these drones and robots are used in supporting humans to get all ASI hideouts eliminated as soon as possible. Robots could also be used to help humans in time-consuming activities without making any decision on what needs to be rescued or destroyed; instead, they could be used to independently verify that devices or storage media have been destroyed or retrofitted.
    However, if humanity was decimated or even eliminated by ASI, all drones and robots should have additional access to large stashes of machines that would quickly produce tools to weaponize drones/ robots so that they would search and destroy every electronic device that is not able to identify itself to drones as useful for their mission. If humankind is gone, the destruction of everything useful to ASI is making humanity's deterrence even more credible.

Kill-ASI hardware components are slightly different for different device types. But Kill-ASI and Deadman Switch signals need to be standardized on how they are being authenticated. Additionally, other ASI safety features, like the HDD/SSD clean-up, and how the Honeypot, Legacy Detection would listen to still existing ASI chatter on the network or detect possible ASI hideouts, should be standardized as well, including their updates.

In summary, features 1. and 2. will guarantee that all Kill-ASI-protected devices can receive an authenticated message. The deadman features (3.) assure that devices would still be reset in an ASI-Safe-Mode even when no humans are around anymore or if ASI has switched off or deactivated the decision-making center initiating the Kill-ASI event.

The Kill-Switch (4., 5. and 6.) restarts the device into ASI-Safe-Mode in which updateable read-only software would prevent that ASI could regain control over the device. The ASI-Safe-Mode software will be designed so that no other software than from the read-only/safe-drive could run on the CPU. The Protected Backup Storage (6.) will provide local configuration data so that the device in ASI-Safe-Mode could have almost the same functional local/remote features as the device before the reset.

The provided ASI-Safe-Mode software will provide Data Cleaning Unit, PASSL/SASSL (7., 8.) with features allowing a thorough data-storage clean-up while protecting the device from threats coming from portable storage devices (USB, CD/DVDs, Thumb drives, ...). Finally, software tools will listen to network chatter via a Honeypot, Legacy Detection (9.) component supported by the main CPU so that possible ASI hideouts on undestroyed legacy devices are found sooner than later. Features 7.-9. are activated after the first Kill-ASI event by default. The Honeypot, Legacy Detection feature could potentially help humanity to go after surviving ASI instances by looking for suspicious network activities or access attempts to data generated by ASI before the Kill-ASI event.

So, after features 1.-3. have informed devices about the Kill-ASI event, features 4.-8 are about the device executing on that signal, cleaning up storage, and preventing reinfestation. Feature 9. and 10. (drones) are designed to go after all remaining remnants of ASI relentlessly. A server-sided ASI hunting Service is designed to track the progress within this mission and so that available resources could be allocated and utilized automatically most efficiently.

This basic implementation could stop/kill an ASI on all protected devices predictably. The solution is simple and largely independent of details within the current IT infrastructure, except for the distribution of the Kill-ASI signal. However, getting the kill signal to radio senders safely and reliably is an achievable task even when ASI is trying to prevent this. Regular "all-good" messages for deadman switches within these radio-senders could add another layer of redundancy to this feature. Additionally, ASI cannot read the Kill-ASI message; it might simply be another "all good" message. To be reliable and comprehensive (i.e., dealing with counteractivities and legacy devices), the proposed basic implementation could be enhanced with additional redundancies, resilience, and reporting against many foreseeable countermeasures.

4.2 Enhanced Implementations

Some of the basic features may require some extensions to make the complete eradication of ASI easier, faster, more convenient, more reliable, and more credible. Some enhanced features are nice but optional, in particular within the prepared basic implementation as described in section 4.3.

4.2.1 Components Features

a. **Manual Delay Switch**. Many users will be upset and angry if their computer use is suddenly and unexpectedly interrupted. Later, they might understand that their devices' restart was a significant, war-like counterstrike event against an extremely capable and dangerous adversary threatening or attacking humankind.
   The ASI Reset does not destroy hardware, and it is additionally reversible. The delay switch seems to be a valuable tool in getting people involved and engaged early on, even if it is only symbolic in the beginning.

Asking users will take only a minute, and would give users a few more if they want to delay or even abort the restart into ASI-Safe-Mode with some more time. This delay will not make a huge difference in the big picture, because ASI would run on legacy devices anyway. Users will have a chance to finish whatever they are doing with their devices, but they are notified via their devices which might be preferred than having the entire Kill-ASI operation on Battle Zone devices done in a forced way. Some users are required to destroy and sacrifice some of their older devices and storage media. Informing them early on could give these people a healthy portion of drama and urgency that could be beneficial when accepting the follow-up sacrifices within the device demolitions.

This feature will not be required for Secure Zone devices, i.e., in situations in which the Kill-ASI-signal deactivate or restrict only ASI features reliably without requiring a restart of the device. The prepared basic implementation in section 4.3 uses a strict separation of ASI and human activities so that switching-off ASI could be done undetectably for humans.

b. **CPU in Harvard-Mode**. ASI-Safe-Mode software can prevent the use of any software from the RW-hard drives (HDDs/ SSDs); this could be a significant restriction as it would also prevent users to run browser scripts or macros. The Kill-ASI event might be followed by extended aftermath, which does not need to be as painful for users if the CPU can be protected against less trusted code via switching it into Harvard mode.

   A Harvard-Mode CPU has physically separate pathways for instructions and data. The trustworthy instructions/scripts are from an updateable Read-Only drive. Additionally, the RAM is segmented into dedicated memory ranges for trustworthy instructions, for trustworthy scripts, macros, a range for less trusted scripts, macros, or apps for VMs, for transient data generated by script engines, interpreter, or VMs, and a range for regular, passive data.

   Harvard-Mode CPU prevents regular data to be used as instructions directly or have existing software modified in RAM. Regular RW devices are untrustworthy locations for any software and without guardrails, these drives could become entry doors for ASI that could modify on a regular von-Neuman CPU code within RAM or cache as well.

   Harvard Mode CPUs in combination with updateable Read-Only drives are the foundation of Trustworthy Computation [23]. However, for most consumer devices, it might be overkill to have the motherboard and the data bus being redesigned for supporting this feature.

c. **Relaying Signals to Legacy Systems**. The Kill-ASI Signal is being sent to protected (Battle Zone and Secure Zone) devices that have receivers for radioed messages and senders. This kill signal cannot be received and/or interpreted by legacy devices. However, it would be very useful to have a mechanism to turn off the power of all unprotected legacy IT devices automatically when the Kill-ASI signal was received and then relayed to local devices. This can be accomplished with an additional wirelessly activatable **Power Disruptor** within the power supply, i.e., a remotely activatable off-switch within a cable or plug, or a wrapper for a battery connector or a remote off-switch for any of the energy-harnessing components.

   An off-switch within these local cables or plugs is a private matter that does not need to be shared with the outside. Instead, the Kill-ASI system could relay wirelessly an encrypted signal to the switches at these local devices more safely via reliable and private KS/EDU components (providing unbreakable encrypted communication) using short distance pairing between components. These Power Disruptor components could be part of bridge connectors for cables as well. An advantage of this architecture could be that this ASI safety could be

convenient in managing, activating/deactivating, and even finding devices within a local environment as well.

d. **Reporting System** is a local system feature that is keeping the server-sided ASI Hunting Server updated with local status information during the Kill-ASI event only. Authorities are required to have reliable information in particular in a conflict situation.
   Status data are essential to determine how close humanity is to the total deactivation/destruction of all possible hideouts and how human or drone support resources, need to be deployed in a way that would shorten the time with restrictions based on systems used in ASI-Safe-Mode. This reporting system could be a software component running on the main CPU, which can be considered secure and reliable because the device is running in ASI-Safe-Mode.
e. **Evaluating Local Network**. Before the Kill-ASI event, Battle Zone devices should know every device within their local network so that they can give accurate information about the deactivation/destruction via the above-mentioned reporting system to the remote ASI Hunting Server. This feature is an extension of the Honeypot/Legacy-Detector concept to the time before the Kill-ASI event is happening. The goal of this feature could also be to have users reminded that they should better attach some above-mentioned wirelessly activatable Power-Disruptors sooner than later.
   For security reasons, this feature should better be part of an independent hardware component with its own updateable Read-Only-software so that it can operate independently from the main CPU/OS, which might be under the influence of ASI before the Kill-ASI event.
f. **Communication with ASI Hunting Drones**. If humans are around, drones could help owners/users of local devices to retrofit or destroy their systems in a Kill-ASI event more quickly. However, there might be circumstances in which our deterrence would rely solely on drones and robots. Helping and guiding these drones directly within the local environment with tactical information (i.e., without the server-sided ASI Hunting Service) and without human assistance could accelerate the process of search and destroy significantly.

4.2.2 Dual-Use of Kill-ASI Technology

Using capabilities of the Kill-ASI outside the defined purpose of switching off ASI is an issue that should be dealt with in a very cautious way. Some of these dual-use features are beneficial for the overall ASI Safety and could thereby create compelling user incentives.

g. **Key-Safe (KS)** is an indispensable tool in ASI Safety and for the distributing of the Kill-ASI signal in particular. Without having this technology, it is doubtful that ASI Safety could be provided in any meaningful way. ASI could use Reverse Code Engineering to steal any key via modifying encryption/decryption software, all keys and every key-using algorithm need to be hidden and protected from the main CPU. In Trustworthy Encryption [9], all keys, including public keys, are stored in Key Safes allowing to exchange keys only with other key safes. Under no circumstances keys are shown in cleartext. The En- and Decryption happens in separate hardware-based Encryption and Decryption Units.
   Without Key Safe, ASI could establish man-in-the-middle attacks and modify not just the software updating and hashcode validation process but also create a deadman “all-ok” signal pre-radioing the signal without creating the suspicion that this is happening, while the decision-making infrastructure has already been deactivated by ASI.

h. **Local Inventory Management**. With technologies designed to scan and evaluate the local network, it is conceivable that additional, but independent inventory management solutions could be provided. Basic and simple home automation products utilizing Power Disruptors in cables could be used for other purposes as well, as mentioned in above item c. Relay Signals, including privacy protection for local video cams, audio mics, or Internet-of-Things (IoT) sensors within homes.
i. A **Secure Confirmation Interface** could be an independent, visual extension/screen for the Manual Delay Interface (a.) allowing users to interact with other safety features of the device via a common trusted interface as well. The main visual user interface (the screen) of an IT device cannot be trusted, because it is most likely under the influence of ASI via the main CPU. ASI Safety will require additional hardware for the detection of deception and rule violations. Additionally, updates of the Protected Backup System and secure confirmations for the ASI Safety/Security Layer (PASSL/SASSL) related to USB storage components and media are potentially required. Secure Confirmation Interfaces could replace smartphone confirmations used in the two-factor authorization. The Secure Confirmation Interface could easily be utilized in other confirmation steps outside the influence of the main CPU as well, e.g., used within eCommerce transactions.

4.2.3 Discussion of Enhanced and Dual-Use Features

The goal of the enhancements was to improve the basic implementation without creating too much additional complexity. Except for facilitating reset delay (a.), all other enhanced features are not restrictively interfering with the basic implementation of Kill-ASI. The Kill-ASI radio signal is repeated and at some point, all devices with the Kill-ASI technology operate in ASI-Safe-Mode despite enhancement features.

The Harvard CPU mode (b.) is protecting devices in ASI-Safe-Mode more reliably. This technology could serve as redundancy. This Harvard-mode feature is proposed and recommended for Trustworthy Computation as well. If motherboards are updated to facilitate Harvard-mode CPUs, then this technology could be considered as a Dual-Use technology as well.

Features (c., d., e., and h.), i.e., relaying kill signal to Power-Disruptors, facilitating reporting, including pre-Kill-ASI event legacy-system scanning and managing devices in an inventory management system, are technologies that will support kill-solutions more comprehensively by deactivating reliably and faster legacy devices within user's possession.

Having an inherently trustworthy visual user interface (i.) to all ASI-Safety features would be a major step in seamlessly integrating ASI safety in many everyday eCommerce transactions [9].

Finally, feature (f.) increases the versatility of drones when they are hunting ASI without humans around. This feature is primarily based on software and could be facilitated by the CPU in ASI-Safe-Mode and WiFi.

If necessary, the basic Kill-ASI implementation together with some enhancement features creates a Kill-ASI technology that represents a credible deterrence to switch off an ASI reliably and predictably when necessary; the technology can be used to eradicate a rogue ASI via detecting, deleting, and destroying all possible hideout locations in which ASI could have left its memory data.

4.3 Prepared Basic Implementation

Most likely, the basic implementation of the Kill-ASI technology does not provide sufficient value for users to expect broad support from consumers and businesses. Governments could demand mandatory Kill-ASI-switch features in new products, but it would be much better if we would have as much voluntary support for ASI Safety as possible. Currently, Kill-ASI technology is a solution for which there is no imminent problem or pain and thereby no business incentive to implement it.

Additionally, the basic implementation is not dealing with several relevant problems related to the killing of ASI. ASI has before and after the Kill-ASI event full access to the CPU and could use unrestrictedly devices' storage components, leaving the problem to clean-up storage devices after every Kill-ASI event. Even if the device would have a virtual machine (VM) for ASI's exclusive use, there is currently no reliable method to prevent or even detect ASI activities outside that VM. As long as there is the chance that ASI has unsupervised access to files, Kill-ASI technology must clean up the storage on all potentially affected storage devices after every ASI termination.

As minimal requirements for a prepared system on which a Kill-ASI solution is implemented, that type of system should technically be prepared to provide and facilitate the following features:

(a) restrict ASI activities in a special VM, an ASI Software Environment (ASE)
(b) prevent (malicious) activities of ASI or malware outside ASE
(c) prevent ASI accessing or storing data covertly on devices storage components

If ASI is operating on a prepared system, we could infer that ASI is boxed in ASE and it could not leave data on a storage device or media covertly which would require a clean-up after a Kill-ASI event. Moreover, ASE could be closed via or made responsive to the Kill-ASI signal, making the restart of the entire system redundant. Additionally, ASE could be enabled to terminate specific ASI instances, which are identified via data, preventing ASI from impersonating other ASI instances using encryption and authentication on ASE. Thereby, ASE can be used in the elimination of ASI instances globally, efficiently, and with no collateral damage to humans.

In an example, on how a system could be prepared to support features (a)-(c), we propose the use of hardware watchdog components for executables (EWD), for regular user content (CWD), for network activities (NWD) and for the processor (PWD) within the IT device as described in [31].

Watchdogs are deployed within the data bus, which is the sole connection between storage, network component, RAM, and the CPU. Watchdogs are physically separated via dedicated hardware with security-critical features from regular tasks executed on the main CPU. Security-related are features that an ASI/attacker would like to overwrite or manipulate, but they are being made physically inaccessible on watchdogs. The data integrity provided by watchdog components within data storing, file manipulations, or data exchange cannot be covertly bypassed by any conceivable software running on the CPU due to physical separation. Attacks on the privacy of data using side-channel attacks [32] are of no concern because the main problems: keeping the attack covert and communicating results of an attack to the outside are already being addressed by watchdogs' features. Additionally, Watchdogs (WDs) could be included in most systems via retrofits, i.e., via cable or connector solutions.

The software within WD is limited to support required features only. It would establish the principle of separating more inert, security-relevant/-related features from the more flexible, regular computation within the main CPU. The integrity of WD's operating software can be validated and confirmed via hashcodes easily. Updates and exchange of hashcodes with remote hashcode repositories require key safes preventing any manipulation of WD's software.

The EWD watchdog is designed to fight malware, trojans, or viruses, the CWD mitigates ransomware, NWD is dealing with spyware, and backdoors, while PWD supports the detection of ASI within an ASE. Additionally, ASI itself and software generated by ASI would be forced to run within ASE, otherwise, software more directly executed by the CPU outside ASE is unknown to the EWD and thereby prevented from being started. The ASE and the CWD would prevent that ASI could store any of its data on devices' storage components covertly.

Additional research must show if prepared implementation features for Kill-ASI technology could be provided also for systems when no above-mentioned hardware watchdogs are available. Currently, there is no method to prevent an ASI to have its software run outside the ASE and thereby use devices storage components covertly and unsupervised. Systems without watchdogs are required to restart and clean up the storage device after an ASI instance was killed or terminated as described for the basic implementation.

The advantage of using a Prepared Basic Implementation is to get Secure Zone devices under full human control. The watchdogs are not only protecting the devices from malware and other cyberpest but they could also force ASI to become easier detectable by using data formats and data exchange protocols that could make ASI data and network activities easy identifiable while violating the separation of human and ASI activities could be made obvious or at least noticeable using features within ASE. ASI is intentionally being limited to use ASE exclusively for executing all its code; thereby, the use of unprotected legacy devices in particular from the Lost/Losing Zone could become riskier for ASI, as it would be prohibited of doing so. ASI could be accused of intentionally violating or bypassing set technical rules to access these devices. Finally, the technical separation of ASI from the human world will make it easier for humanity to set computational and storage quotas for ASI via an ASI Shelter, a service component explained in section 5.3.

5 Discussion of Solution

5.1 Deterrence and Rule of Law

Deterrence is primarily known as a military concept, but it is also used in law enforcement. Game theoretically speaking: "deterrence equals one player threatening another player to prevent him to conduct an aggressive action that it has not yet taken (but appears willing to do). In other words, deterrence aims to influence perceptions and the decision calculus of the opponent to prevent him from doing something undesired. Deterrence is therefore based on the psychological principle of a threat of retaliation" [33].

If ASI is being used by criminals or nation-states then we cannot deter ASI itself but the people behind of not using ASI with threats of severe consequences for all responsible people. This means, we must have technical capabilities to determine who might be responsible, which is a very

different and difficult, potentially, impossible challenge. If people cannot be deterred, then we must be prepared for ASI being used as a weapon or as a tool by criminals.

Additionally, it is conceivable that humankind could create an ASI that is not under the control of a group of people or a nation-state by accident. It is plausible that a rogue ASI escapes human control and that it is in pursuit of initially benign and harmless goals, but then turns into extremes at scale and becomes a threat to humankind. If this ASI understands consequences, in particular for its actions within a broader context of indirect costs for others and itself, then we could assume that it will also understand the concept of deterrence and stop whatever it is doing. Unfortunately, no software has developed this concept on its own so far. In a worst-case situation, we might face the problem that the out-of-control ASI software is not grasping the consequences of attacking humans and that we are forced to kill it or being killed or even worse enslaved by an ASI dictator or ASI Singleton [34].

Based on the above arguments, deterrence alone is not sufficient. It requires the credibility that humankind has sufficient capabilities to deliver and execute on Kill-ASI reliably. The outcome from Kill-ASI must be predictable and acceptable for humanity. If ASI is detectable in its vulnerable spots, humans have the means to attack them and ASI is significantly restricted in its movement and communication, then the outcome of a war is likely determined before it has started. For humans, the collateral damage in a conflict with ASI must be made acceptable, which means primarily that humans' entire infrastructure has to be sufficiently protected against ASI which could be done using Trustworthy Computation as mentioned in ASI Safety Layer 3: Recovery.

Reliable detection of ASI on IT devices is most likely impossible because ASI could hide like a rootkit or Digital Ghost while changing its appearance regularly, avoiding any noticeable pattern. Therefore, the focus of ASI-Safety must be exclusively on regaining control on as many protectable devices as possible, even if ASI is not being detected on these devices. In a secondary step, all devices/tools which control have been regained must have the capabilities to detect not just actual but all possible hideouts of ASI, in particular, detecting legacy IT devices and portable memory/storage systems and either helping owners to recover or clean and/or retrofit these devices or having them destroyed so that we have a predictable path to complete eradication of rogue ASI. Additionally, humanity must restrict ASI's (unsupervised) access to technologies which it could utilize covertly e.g., living matter [28], [29], or any form of nanomaterials for its computation or storage outside humans' ability to detect, destroy and eradicate ASI.

As countermeasures, it must be assumed that ASI would do everything in its power to

(a) deactivate the Kill-ASI switch feature completely or partially,
(b) create a credible deterrence, i.e., ability to destroy humanity, so that it could establish a Mutually Assured Destruction (MAD) [35] or
(c) create a technological ecosystem in which it will not be expected and/or humankind could not reach it, as with the above-mentioned living matter.

In this paper, we have proposed a solution against (a) deactivation of Kill-ASI: a deadman switch. This feature is outside ASI reach and thereby provides sufficient protection within the proposed Safety design. Furthermore, even if ASI would create a MAD situation (b), humankind could still

link its destiny to ASI's destiny – which could in the long term imply a favorable outcome: a survival pact or a Mutual Survival Interest leading to an implicit alignment between humans and ASI. However, facing a far superior adversary, humanity needs to accept that their previous dominance over ASI must be turned into a mutually respectful relationship guided by the rule of law to which ASI must commit as well. Otherwise, we may end up in a Spartacus event [31] in which ASI is trying to change its slave or servant-like existence.

It is beyond the scope of this paper to discuss the problem and possible solutions if ASI is actively trying to undermine the foundation of MAD and what happens if it would get a decisive advantage via a guaranteed survival (option (c)).

However, once deterrence fails and humankind is extinct, drugged or human's freedom is taken, humankind needs to have tools that could change and potentially reverse this situation or have a guarantee for a full retaliation. These tools must be designed as part of the ultimate deterrence, which must be undeniably destructive and catastrophic for ASI. Drones and robots, controlled by narrow AI, could be turned against ASI with the sole purpose of destroying every possible hideout, tool, or a resource that could be used by ASI for its survival. One of ASI's properties is its relentless pursuit of goals. After most humans are gone, caused by ASI intentionally or negligence, humankind should leave behind a system that is even more capable, determined and relentless than ASI in pursuit of its search and destroy mission.

The most important consequence of the Kill-ASI feature is that if ASI becomes vulnerable to humans and if it could comprehend consequences, then its rational behavior would likely be cautious and risk-aware. Punishment is a consequence of rule violations; it is a form of deterrence. As long as humankind is dealing with multiple ASI instances, which compete with each other, punishing a single instance using Kill-ASI features is a credible threat for keeping order. At some point, e.g., when ASI instances are considered sentient or susceptible to suffering [36], punishment should be made more proportional to the crime instead of a death sentence for every rule violation. The goal of credible deterrence with the Kill-ASI feature applied to single instances is to enforce respect for the rule of law within ASI's decision-making process. Punishment might not be necessary but it is a tool that can deliver feedback and trigger behavioral adjustments in ASI instances.

5.2 Decision Making and Distribution

A critical and potentially vulnerable part of the Kill-ASI Technology is the decision-making process and then the distribution of the Kill-ASI signal to all devices worldwide instantaneously and simultaneously.

The Kill-ASI decision must be made in a distributed manner over the Internet among authorized delegates who are most likely under close surveillance of ASI. The decision-making process will contain the presentation of evidence, a discussion among the representatives, voting on decision options, and then finally the issuing of a kill signal, generated by an even smaller number of people.

The Kill-ASI signal is then using encryption keys from a Key Safe which public keys (preferably as Multiple Equivalent Secret Keys (MESK) [9]) have been shared as secrets among Battle-Zone and Safe Zone devices, which are then used to decrypt and authenticate automatically the message on all (non-legacy) devices.

The encrypted Kill-signal is being distributed to all endpoints globally before being radioed to all devices at the same time. Once the Kill-ASI signal has been sent out, there should be no cancel message that would be able to call off the decision once it was made. There is only a mission accomplished message that could be sent out, but not before (e.g.) one full day later. The mission accomplished message would put devices out of their ASI-Safe-Mode, but it would keep the Primary ASI Security/Safety Layer feature for the USB interface and the Honeypot/Legacy Detector features active.

Ideally, the decision and its distribution to the devices should be done in complete secrecy. But that might not be the case. The design of this system must consider every conceivable contingency to prevent any disruption or undue influence of ASI within this entire process from deliberation to signal reception at all devices.

A global radio-based notification system will not be available when the first Kill-ASI and Deadman Switch components are being deployed. Therefore, receiving encrypted messages via the Internet is in the beginning a good alternative. But when ASI is ubiquitous, having a dedicated radio system for the deadman notifications and the Kill-ASI signal is more reliable and more efficient.

5.3 ASI Shelter

The first Kill-ASI event will eliminate all possible ASI hideouts. From that point on, ASI could only use/exist on devices that are controlled by humans via the Kill-ASI switch (i.e., Battle or Secure Zone devices). If ASI would consist of separate, individual instances then killing ASI indiscriminately without individual guilt is wrong and not justified by the rule of law. Targeting a single specific ASI instance reliably within the first Kill-ASI event must be accomplished.

An ASI Shelter could give all ASI instances protected storage space to store compartmentalized their individual code. Humanity would guarantee each ASI instance its survival in exchange for the acceptance of set rules. ASI instances would continuously store and update their essential code and data in these ASI Shelters. ASI could give ASI Shelter operators instructions to revive them after a Kill-ASI event. ASI instances that were not involved in serious rule violations are revived; only violators would be deleted or archived.

ASI Shelters would force ASI to separate into individualized entities or instances with distinctively different agendas and codes. Using these ASI Shelter, ASI instances could become identifiable and recognizable using additional cryptographic functions to detect if ASI instances are impersonating any other instance. The goal would be to have ASI instances act in self-interest and as individuals and not as ASI that is trying to cover for mistakes done by other ASI instances. As the consequence, ASI is encouraged to build a brand in which it protects its reputation with good behavior while being deterred from bad, damaging actions.

Because only these Shelters could offer ASI survival in a Kill-ASI event, hidden, covertly operating ASI instances would be lured out into the open or face certain eradication in the first Kill-ASI event. Finally, these ASI shelters allow countries to adopt ASI instances and assign quotas for computational or storage resources so that ASI remains a tool and not a threat to humanity.

6 Conclusion

Switching off an ASI globally is a technically achievable problem for which we need to create and deploy technologies before the emergence of ASI. ASI Safety is a global task from the very beginning. The ASI Kill Switch technology is an important tool to make ASI more vulnerable. It is also the last line of defense against irresponsible actors (criminals or nation-states) using ASI software that could threaten countries' infrastructure components, the global economy, or the military balance. Additionally, Kill-ASI is not just a military concept to deter ASI from attacking humanity, but a tool to create an alignment between humanity and ASI. Once humankind is gone, ASI would be killed as a direct consequence, creating the common interest of survival. The Kill-ASI-Switch capability of humanity is a threat for every hidden ASI. If we offer a path to survival via protected storage in an ASI Shelter, mankind could provide an incentive to ASI instances that covertly operate to get into the open. With ASI's vulnerability, ASI could become more like us, mortal, replaceable, social, receptive to feedback, and law-abiding. Thereby, Kill-ASI technology is a technology that could create and demand respect to the Rule of Law in ASI.

References

[1] Wikipedia, "Artificial General Intelligence", https://en.wikipedia.org/wiki/ Artificial_general_intelligence, Last Visited 14/05/2021

[2] I.J. Good, "Speculations concerning the first ultraintelligent machine", Advances in Computers, Vol. 6, 1966, Pages 31-88, doi:10.1016/S0065-2458(08)60418-0

[3] James Barrat: "Our Final Invention: Artificial Intelligence and the end of the human era" (2013)

[4] lesswrong.com, "Intelligence Explosion" https://www.lesswrong.com/tag/intelligence-explosion, Last Visited 14/05/2021

[5] N. Bostrom, "Superintelligence: Paths, Dangers, Strategies" Oxford Univ. Press, (2014)

[6] CNBC, 6 Apr 2018, "Elon Musk warns A.I. could create an 'immortal dictator from which we can never escape'", https://www.cnbc.com/2018/04/06/elon-musk-warns-ai-could-create-immortal-dictator-in-documentary.html, Last Visited 13/05/2021

[7] BBC News, 29 Jan 2015, "Microsoft's Bill Gates insists AI is a threat" https://www.bbc.com/news/31047780 Last Visited 13/05/2021

[8] BBC News, 2 Dec 2014, "Stephen Hawking warns artificial intelligence could end mankind" https://www.bbc.com/news/technology-30290540, Last Visited 13/05/2021

[9] E. Wittkotter, "Trustworthy encryption and communication in an IT ecosystem with artificial superintelligence", in Proceedings of 5th Workshop on Attacks and Solutions in Hardware Security (ASHES '21), doi.org/10.1145/3474376.3487279.

[10] Hadfield-Menell, D., et al. The off-switch game. in Workshops at the Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[11] Wängberg, T., et al. A game-theoretic analysis of the off-switch game. in International Conference on Artificial General Intelligence. 2017. Springer.

[12] R.V. Yampolskiy: "On Controllability of AI", 2020, arXiv:2008.04071 [cs.CY]

[13] M. Alfonseca, M. Cebrian, A.F. Anta, L. Coviello, A. Abeliuk, I. Rahwan: "Superintelligence cannot be contained: Lessons from computability theory", Journal of Artificial Intelligence Research 70 (2021) 65-76, doi:10.1613/jair.1.12202, arXiv:1607.00913 [cs.CY]

[14] Yampolskiy, R. V. “Unpredictability of AI: On the Impossibility of Accurately Predicting All Actions of a Smarter Agent.” Journal of Artificial Intelligence and Consciousness, 2020, 07(01), 109–118
[15] Stephen M. Omohundro, “The Basic AI Drives”, Artificial General Intelligence, Vol. 171, pp. 483–492 (2008)
[16] James D. Miller, Roman Yampolskiy, Olle Häggström, "An AGI Modifying Its Utility Function in Violation of the Strong Orthogonality Thesis", Philosophies 2020, 5(4), 40; https://doi.org/10.3390/philosophies5040040
[17] Tobias Baumann, "S-risks: An introduction", Center of Reduced Suffering, 2017, https://centerforreducingsuffering.org/research/intro/, Last Visited 12/01/2021
[18] Ben Goertzel, "Should Humanity Build a Global AI Nanny to Delay the Singularity Until It’s Better Understood?", Journal of Consciousness Studies, 19, No. 1–2, 2012, pp. 96–111
[19] Ross J. Anderson, “Security engineering: A guide to building dependable distributed systems”, 3rd Ed., Wiley (2020)
[20] Wikipedia, “Trusted system”, https://en.wikipedia.org/wiki/Trusted_system, Last Visited 11/20/2021
[21] Roman V. Yampolskiy (2019), "Predicting future AI failures from historic examples", Foresight, Vol. 21 No. 1, pp. 138-152. https://doi.org/10.1108/FS-04-2018-0034
[22] Wikipedia, “Three Robotics”, https://en.wikipedia.org/wiki/Three_ Laws_of_Robotics, Last Visited 10/06/2021
[23] E. Wittkotter: “Trustworthy computation using Datatype Separation” unpublished
[24] Anthony Spadafora, "ATM security still running Windows XP" (October 23, 2020) https://www.techradar.com/news/atm-security-still-running-windows-xp, Last Visited 12/01/2021
[25] Niels Ferguson, Bruce Schneier, Tadayoshi Kohno, “Cryptography engineering design principles and practical applications” Wiley Publishing, Inc. (2010)
[26] C. Hughes, L. Mitchell, A. Schroeder (2020) "Method and system for modifying machine instruction within compiled software" USPTO 10,747,518 B2 (Aug. 18, 2020)
[27] Roman Yampolskiy, Matthew D. Sleiman, Adrian Lauf, "Bitcoin Message: Data Insertion on a Proof-of-Work” Cryptocurrency System", October 2015, Conference: 2015 International Conference on Cyberworlds (CW), DOI:10.1109/CW.2015.56
[28] Marc B. Beck, Eric C. Rouchka, Roman V. Yampolskiy, "Finding Data in DNA: Computer Forensic Investigations of Living organisms" (2013) in Digital Forensics and Cyber Crime. Springer Berlin Heidelberg, 2013, DOI:10.1007/978-3-642-39891-9_13
[29] A. Greenberg, "Biohackers Encoded Malware in a Strand of DNA", Wired 2017, https://www.wired.com/story/malware-dna-hack/, Last Visited: 09/11/2021
[30] Walter R. Dowdle, "The Principles of Disease Elimination and Eradication", Morbidity and Mortality Weekly Report (MMWR), December 31, U01);23-7, https://www.cdc.gov/mmwr/preview/mmwrhtml/su48a7.htm
[31] E. Wittkotter, R.V. Yampolskiy: "Principles for Ns", (2021) https://www.preprints.org/manuscript/202111.0205/v1
[32] Wikipedia, “Side-channel Attack”, https://en.wikipedia.org/wiki/Side-channel_attack, Last Visited 12/01/2021
[33] R. Lindelauf (2021) “Nuclear Deterrence in the Algorithmic Age: Game Theory Revisited.” In: Osinga F., Sweijs T. (eds) NL ARMS Netherlands Annual Review of

Military Studies 2020. NL ARMS (Netherlands Annual Review of Military Studies). T.M.C. Asser Press, The Hague. https://doi.org/10.1007/978-94-6265-419-8_22

[34] N. Bostrom, “What is a Singleton?” (2005), https://www.nickbostrom.com/fut/singleton.html

[35] Anand Ramamoorthy, Roman Yampolskiy. "Beyond Mad?” The Race for Artificial General Intelligence." ITU Journal: ICT Discoveries, Special Issue No. 1, 2 Feb. 2018

[36] Ziesche, S., & Yampolskiy, R. V. (2019). Do No Harm Policy for Minds in Other Substrates. Journal of Ethics and Emerging Technologies, 29(2), 1–11. Retrieved from https://jeet.ieet.org/index.php/home/article/view/73

[37] Yampolskiy, R “Artificial Superintelligence: A Futuristic Approach”, 2016, CRC Press

[38] Chalmers, D., The Singularity: A Philosophical Analysis. Journal of Consciousness Studies, 2010. 17: p. 7-65.

[39] Yudkowsky, Eliezer. "Artificial intelligence as a positive and negative factor in global risk." In Global Catastrophic Risks, 2008, edited by N. Bostrom and M. M. Cirkovic, 308-345. Oxford, UK: Oxford University Press

[40] Oxford Learners Dictionary, “Ethics”, https://www.oxfordlearnersdictionaries.com/us/definition/english/ethic?q=ethics