Artificial Intelligence and Social Computing, Vol. 28, 2022, 59–66
https://doi.org/10.54941/ahfe1001447

The Effect of Varying Levels of Automation During Initial Triage of Intrusion Detection

Daniel N. Cassenti[1], Aayushi Roy[2], Thom Hawkins[3], and Robert Thomson[4]

[1]DEVCOM Army Research Laboratory Adelphi, MD 20783, USA
[2]University of Maryland College Park, MD 20742, USA
[3]PM Mission Command Aberdeen Proving Ground, MD 21005, USA
[4]Army Cyber Institute, United States Military Academy West Point, NY 10996, USA

ABSTRACT

With unrestrained optimism regarding the possibilities of artificial intelligence (AI) exceeding its actualization, AI developers are under increasing pressure to integrate AI into complex human decision-making tasks without fully understanding the implications of this automation. To investigate how automation may influence human performance in a high workload environment, this study utilizes a triage scenario from intrusion detection using a simulated SNORT interface. Participants classify a series of time-sensitive alerts as real intrusions or false alarms with the assistance of varying levels of automation (LOA) from no automation to fully autonomous. Preliminary results showed that participants tend to prefer and have some performance benefits with intermediate levels of automation.

Keywords: Intrusion detection, Levels of automation, Human-AI collaboration

BACKGROUND

Cyber analysts are inundated with the exhausting task of manually reviewing tens of thousands of network intrusion alerts per day, with nearly all of them being false alarms (Thomson, 2018). Their task involves quickly whitelisting as many of the false alarms as possible and passing possible threats up the chain for further analysis. Despite being tedious and having a relatively low hit rate, the cost of missing a true alert is severe: an adversary may gain access to your secured network. From that point, it can take upwards of 200 or more days to detect such intrusions after they are initially missed. Compounding this problem of alert fatigue, the Prevalence Paradox (Sawyer and Hancock, 2018) identifies that even under ideal conditions, when true hit rates fall below one percent there is an inadvertent drop in vigilance to detect the true alert signal.

This high false-alarm rate is due to the fact that much of intrusion detection involves anomaly detection over network packets and computer processes, usually involving rules that only look at a single point in time. For instance, a rule looking for possible ransomware by flagging a large quantity of hard drive writes could falsely flag a computer zipping up a dataset to be archived. To combat this issue, many researchers are turning to automation techniques to process many of these alerts for the analysts.

With a rush to provide technical advancements to analysts by developing novel automation, there has been a scarcity of research into the effects of varying levels of automation in the cyber domain. A review of research into explainable artificial intelligence has shown that adding automation may actually increase workload (to understand the automation) while not necessarily increasing performance (Endsley and Kaber, 1999). Furthermore, if the human does not trust the AI appropriately, then the AI may be over- or under-used based on its actual competency. AI-based automation that is too intrusive may overwhelm the human with added workload, especially if the interface has not been adequately designed (Sawyer and Hancock, 2018).

Taken together, in the context of cyber analysts any form of automation needs to not only reduce the operator's overall workload and/or increase efficiency (without increasing workload), but also needs to take into account the limitation of human vigilance. In the remainder of this paper we describe how we adapted Sheridan & Verplank's (Sheridan and Verplank, 1978) Levels of Automation (LOA) paradigm into an intrusion detection scenario, as well as provide some preliminary results as we validate our methodology.

LEVELS OF AUTOMATION

Operators and autonomous systems each have relative advantages and disadvantages that interact in a manner more complex than the sum of each of their abilities. Sheridan and Verplank (Sheridan and Verplank, 1978) initially investigated varying LOA for teleoperation of an undersea explorer robot. They identified that optimal LOA should be decided on a case-by-case basis based on task demands and the capabilities of the automation in question, and described a set of ten levels from no automation through fully-autonomous. For an autonomous system to be used effectively, a user must believe it to be capable of performing well, especially when human decision-making begins to be automated by the system. If the user does not trust the system, it will not be used. The extant body of literature (Endsley and Kaber, 1999) identifies that an intermediate level of automation is generally appropriate for most dynamic control tasks roughly analogous to intrusion detection. Specifically, this is when automation makes recommendations that an operator can accept or override.

Endsley and Kaber (Sheridan and Verplank, 1978; Kaber and Endsley, 2004) argue that a subset of five-levels is best applicable to human-automation tasks where an operator/analyst needs to make decisions based on system information. Table 1 represents the levels of this model in ascending order of proportion of tasks performed by automation.

In determining the optimal LOA for cyber analysts performing initial triage, we are focusing primarily on two factors: the effectiveness of the automation and the operator's level of trust in the automation. Other factors (such as overall analyst workload and design decisions in the user interface) are

Table 1. Levels of automation from Endsley (Kaber and Endsley, 2004).

Level	Name	Task could be performed by the:
1	Manual Control	Operator with no assistance from the system
2	Decision Support	Operator with input in the form of recommendations provided by the system
3	Consensual AI	System with the consent of the operator required to carry out actions
4	Monitored AI	System to be automatically implemented unless vetoed by the operator
5	Full Automaton	System with no operator interaction

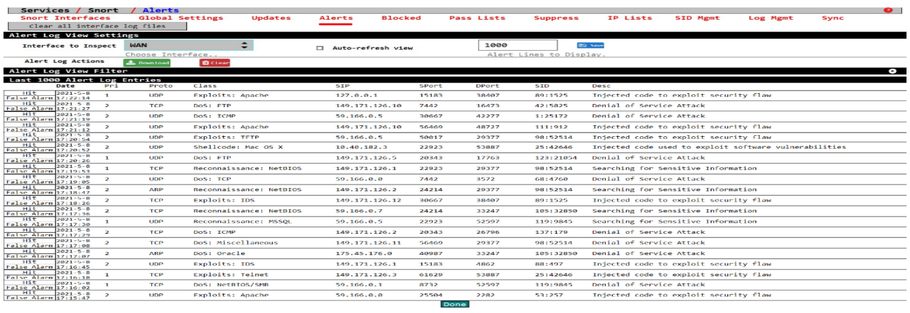
beyond the scope of the present study, although we acknowledge that they will bias the analyst, however those effects should be indirectly influenced by their propensity to trust the automation.

Previous research (Onnasch et al. 2014; Wickens, 2002) has clearly shown a threshold LOA beyond which negative consequences result and increase in degree of severity with increasing LOA. This threshold is characterized as the point at which automation moves from information processing functions to action selection (i.e., moving from LOA3 to LOA4 in the present study), and this is due to a complex set of psychological factors in the analyst. While offloading part of a complex task to a teammate can help overall task performance, there can be negative consequences when an analyst believes that the teammate is not as effective or does not think similarly to themselves. The intensity of the negative effects of teaming is exacerbated when the collaborator is automated (Onnasch et al. 2014). As pointed out by (Onnasch et al. 2014), it is exceedingly difficult to consider AI as a teammate. Furthermore, many automation algorithms are black box (Thomson and Schoenherr, 2004), that is, one cannot explain how the automation is making a given decision.

INTRUSION DETECTION METHODOLOGY

To establish the impacts on varying levels of automation during initial triage of intrusions detection, we must first identify a scenario with sufficient ecological validity to have a path to influence operational environments, but also be amenable to prototyping on non-expert users. We chose to simulate the operation of the Snort interface (Caswell and Beale, 2004) (see Figure 1) as it is a real Intrusion Detection Software (IDS) which utilizes rule-based network traffic alerts to present to a given analyst. While the full task of the analyst would involve additional visualization of network traffic based on temporal patterns, the Snort interface provides a reasonably complex simulation environment to validate against novice users.

Cyber analysts work long hours on a tedious task, which compounds the chances of having reduced vigilance and is further exacerbated by the high false alert rate. Analysts view alerts in software that organizes alert elements into categories in a spreadsheet-like view. The rule-based IDS operates on all



Interface to Inspect: WAN (Choose Interface..)
Auto-refresh view
1000 (Alert Lines to Display.)
Alert Log Actions

Last 1000 Alert Log Entries

	Date	Pri	Proto	Class	SIP	SPort	DPort	SID	Desc
Hit False Alarm	2021-5-8 17:22:14	1	UDP	Exploits: Apache	127.0.0.1	15183	38407	89:1525	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:21:27	2	TCP	DoS: FTP	149.171.126.10	7442	16473	42:5825	Denial of Service Attack
Hit False Alarm	2021-5-8 17:21:19	2	UDP	DoS: ICMP	59.166.0.5	30667	42277	1:25172	Denial of Service Attack
Hit False Alarm	2021-5-8 17:21:12	2	UDP	Exploits: Apache	149.171.126.10	56469	48727	111:912	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:20:54	2	UDP	Exploits: TFTP	59.166.0.5	50017	29377	98:52514	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:20:52	2	UDP	Shellcode: Mac OS X	10.40.182.3	22923	53887	25:42646	Injected code used to exploit software vulnerabilities
Hit False Alarm	2021-5-8 17:20:26	1	UDP	DoS: FTP	149.171.126.5	20343	17763	123:21054	Denial of Service Attack
Hit False Alarm	2021-5-8 17:19:53	1	TCP	Reconnaissance: NetBIOS	149.171.126.1	22923	29377	98:52514	Searching for Sensitive Information
Hit False Alarm	2021-5-8 17:19:05	2	UDP	DoS: TCP	59.166.0.0	7442	3572	68:4760	Denial of Service Attack
Hit False Alarm	2021-5-8 17:18:47	2	ARP	Reconnaissance: NetBIOS	149.171.126.2	24214	29377	98:52514	Searching for Sensitive Information
Hit False Alarm	2021-5-8 17:18:26	2	TCP	Exploits: IDS	149.171.126.12	30667	38407	89:1525	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:17:36	2	TCP	Reconnaissance: NetBIOS	59.166.0.7	24214	33247	105:32850	Searching for Sensitive Information
Hit False Alarm	2021-5-8 17:17:30	1	UDP	Reconnaissance: MSSQL	59.166.0.5	22923	52597	119:9845	Searching for Sensitive Information
Hit False Alarm	2021-5-8 17:17:29	2	TCP	DoS: ICMP	149.171.126.2	20343	26796	137:179	Denial of Service Attack
Hit False Alarm	2021-5-8 17:17:08	2	TCP	DoS: Miscellaneous	149.171.126.11	56469	29377	98:52514	Denial of Service Attack
Hit False Alarm	2021-5-8 17:17:07	2	ARP	DoS: Oracle	175.45.176.0	40987	33247	105:32850	Denial of Service Attack
Hit False Alarm	2021-5-8 17:16:45	2	UDP	Exploits: IDS	149.171.126.1	15183	4862	88:497	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:16:18	1	TCP	Exploits: Telnet	149.171.126.3	61629	53887	25:42646	Injected code to exploit security flaw
Hit False Alarm	2021-5-8 17:16:02	1	TCP	DoS: NetBIOS/SMB	59.166.0.1	8732	52597	119:9845	Denial of Service Attack
Hit False Alarm	2021-5-8 17:15:47	2	UDP	Exploits: Apache	59.166.0.0	25504	2282	53:257	Injected code to exploit security flaw

Figure 1: A mockup of the Snort interface (Hart, 2006).

of the traffic and filters the data, only passing along the traffic that is potentially dangerous. The elements of alerts an analyst scans for to determine whether an alert is benign or threatening covers a wide range of potential warning signs. The suspect elements could be from given IP addresses or classes of IP addresses (indicating a specific attacker), type of activity (e.g., accessing a sensitive file), or outbound traffic from a potential insider threat. Spreading attention over so many different fields increases the cognitive workload demands on the analyst. In addition, analysts need to perform a host of different higher-level cognitive tasks to perform their duties. These include (but are not limited to) pattern recognition, grouping, place keeping, filtering, reading, memory recall, and decision making.

To integrate varying LOA, we created a whitelisting column at the left of the interface where an automated AI may make recommendations (or full decisions) for the user. The user's task is to determine whether a given row's alert is a true threat or a false alarm based on a set of rules provided at the beginning of the scenario.

PRESENT STUDY

A pilot interface was designed based on the Snort interface in Figure 1, integrating an additional response column that was used to provide an AI-based recommendation on whether the alert is a true positive or false alarm. The study examines how well participants will use an automated AI based on five levels of automation from no-automation through full automation, with the critical levels being 2-4 (decision support, consensual AI, and monitored AI, respectively), and varying attack rate to gauge participants' sensitivity to the attack signal. A simplification to cut down on the information that participants are required to process, alerts designated as threats were limited to ones with one of a list of 20 codes that indicated a true threat which were spread evenly across four fields, instead of having participants searching for patterns across multiple alerts.

The complete study was a mixed 5 (between; LOA) x 2 (within; hit rate 5% or 20%) counterbalanced design. The AI was programmed with an 80% hit rate and 20% false alarm rate, whose values were decided-upon based on pilot research.

METHODS

Participants

A total of 78 participants were recruited through Amazon Mechanical Turk. Of those, 23 did not meet device compatibility requirements to run the experiment, 15 participants were bots and dropped, and 40 participants successfully completed the pilot study.

Materials

The study was designed using the E-Prime 3.0 experiment software and participants completed the study using the E-Prime Go 1.0 application. Stimuli were designed by creating a set of five relevant fields derived from the Snort IDS interface. These fields include the Class of alert, the Session Initiation Protocol, the Source Port, and the Destination Port. After the session, participants completed three surveys: NASA-TLX workload (Hart, 2006), Trust in Automation (Chien et al. 2014), and the System Usability Scale (Bangor et al. 2008).

Procedure

Participants initially were instructed to check a series of system requirements to determine whether their systems were compatible with the software. They were then provided $1 and if compatible, provided the option to complete the main study for an additional $7.50.

The experiment began with an initial-instructions screen which described the task as acting as a cyber analyst to correctly identify as many hits and false alarms as they could. The next instructions screen provided the example interface (see Figure 1). The third screen instructed the participants that threats contained one of five prescribed codes in each of four fields. They were encouraged to write these codes down so they could refer to them during the course of the experiment. We encouraged this action because there was no means of verifying that participants would not write the codes down, so we decided to offer all of them the chance to do so. This final instructions screen also informed the participants that they would be scored for each of the 20 alerts in the following way: Threat reported as a hit earned ten points; Threat reported as a false alarm lost ten points; Nonthreat reported as a false alarm earned one point; and Nonthreat reported as a hit lost one point. We believed that this ten-to-one ratio would produce an emphasis on correctly identifying hits. Participants were also instructed that they would have a time limit of 1.5 minutes per trial. They were also instructed that they would earn an extra $2.50 over their $7.50 fee if they got to a certain undisclosed score threshold to promote a focus on accuracy and maintaining vigilance throughout the task. In truth, everyone who performed the experiment received the bonus.

Participants moved from the instructions to one trial of practice at LOA level 1. Each alert had a 20% chance of being a threat. Participants used their mouse to click on the "Hit" button to indicate a threat or the "False Alarm" button to indicate a nonthreat. When they finished, they pressed the

"Done" button. No other buttons responded to clicks on the interface. A feedback screen appeared following the time limit or when the participant pressed "Done" that presented the response and correct answer for each alert. No score appeared for the practice, however in the main experiment, each feedback screen showed the new score and the score from after the previous trial.

Following the practice feedback screen, participants were instructed about the AI that would operate while performing the main trials. The following are summaries of those instructions: LOA 1 AI would operate in the background but not offer help; LOA 2 AI would provide recommendations for each alert, but participants would need to select their own response for each alert (distinguished by color); LOA 3 AI would provide recommendations that participants would merely need to leave unclicked if they agreed with them; LOA 4 AI would provide recommendations, but if the subject overrode one, all the recommendations would disappear and the participant would need to select a response for each alert; and LOA 5 would make all decisions for the participants and skip the response screen to present the feedback screen.

Within a block, participants would see 10 trials. In each trial, 20 alerts were presented on the display and the participant had to mark whether it was a threat or false alarm (in LOA1/2), or to accept/reject the AI recommendations (LOA3/4). Each participant performed two experimental blocks counterbalanced by block order based on even and odd subject number. One block had one threat per 20 alerts (5%), averaged over all trials and the other had four threats per 20 alerts on average (20%). Participants received feedback after each trial with their current score and how many alerts they hit and missed. Detecting or failing to detect a nonthreat led to a respective increase or decrease of one point, while detecting or failing to detect a threat led to a respective increase or decrease of ten points. Following the two blocks of trials, participants completed the surveys and were thanked and reimbursed.

RESULTS & DISCUSSION

A summary of our results for this pilot study is presented in Table 2 below. Accuracy is almost uniformly high; however, this is likely due to a strategy of defaulting to pressing 'false alarm' which would result in 95% accuracy and 80% accuracy respectively between the 5% and 20% conditions. This interpretation was supported by participants' stronger positive response bias. More task appropriate measures would be to examine the related miss and false alarm rates. The AI (see LOA5 for AI performance on Full Automation) exhibited the average 15% miss rate and 20% false alarm rate, while humans tended to miss 40-55% of all true attacks while exhibiting a relatively low false alarm rate between 1-12%. This resulted in participants having slightly higher sensitivity than the AI in LOAs 1-3, despite a greatly higher miss rate.

The results of this study offer a striking comparison between task demands and how well participants integrated with the LOA. Participants across LOAs 2-4 were likely to override the AI recommendation of a Hit, resulting in a much higher miss rate as well as a lower false alarm rate. This was despite

Table 2. Summary of mean participant results for the present study.

Condition	Hit Rate	FA Rate	Response Bias	D'	Overall Accuracy	
LOA1	5%	0.4476	0.0166	1.130557	1.9978	0.9533
	20%	0.4919	0.0089	1.194268	2.3481	0.8912
LOA2	5%	0.4333	0.0116	1.219222	2.1027	0.9584
	20%	0.6439	0.0577	0.602723	1.9435	0.8816
LOA3	5%	0.5100	0.0326	0.909191	1.8685	0.9445
	20%	0.6700	0.0600	0.55743	1.9947	0.8860
LOA4	5%	0.5600	0.1276	0.493393	1.2887	0.8465
	20%	0.5900	0.1013	0.523344	1.5018	0.8361
LOA5	5%	0.8500	0.2070	-0.10981	1.8532	0.7958
	20%	0.8104	0.1969	-0.0133	1.7323	0.8046
Overall	5%	0.5958	0.0909	0.546453	1.5781	0.8913
	20%	0.6671	0.0969	0.433691	1.7310	0.8556

the cost of a false alarm being set to -1 while the cost of a miss was -10, which should have biased them to a risky strategy leading to relatively higher false alarms. This implies that instructions alone were insufficient to convey the relative cost of missing a true intrusion. Finally, participants also exhibited the most trust in automation (3.6/5) in LOA2 (making only recommendations) compared with other conditions (averaging 2.8/5). Whether participants were implicitly aware of the higher false alarm rate or had some other reason for not trusting the AI will be a focus in future work using this interface.

CONCLUSION

The goals of this study were to establish a paradigm in which LOA could be investigated with the purpose of improving the human factors of AI aids in human-AI collaboration tasks, specifically initial triage of intrusion detection. While several interesting results appear to validate our broad hypothesis that an intermediate level of automation best supported human-AI collaboration for intrusion detection, there is much future work which still needs to be completed to determine a broader Receiver-Operator Curve (ROC) for human-AI collaboration by varying the payoff matrices, fine-tuning the automation in LOA4, and wire-framing UI improvements to best present the AI decisions/recommendations to the user.

REFERENCES

A. Bangor, PT Kortum, & JT Miller, "An empirical evaluation of the system usability scale." *Intl. Journal of Human–Computer Interaction,* vol 24, 2008, pp. 574–594.

B. Caswell & J. Beale. "Snort 2.1 intrusion detection," *Elsevier*. 2004.

BD Sawyer & PA Hancock. "Hacking the Human: The Prevalence Paradox in Cybersecurity," *Human Factors,* vol 60, 2018, pp. 597–609.

C. Wickens, "Multiple resources and performance prediction." *Theoretical issues in ergonomics science,* vol 3, 2002, pp. 159-177.

DB Kaber & MR Endsley. "The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task." *Theoretical Issues in Ergonomics Science*, vol 5, pp. 113–153, 2004.

L. Onnasch, CD Wickens, H. Li, & D. Manzey. "Human performance consequences of stages and levels of automation: An integrated meta-analysis." *Human factors* vol 56, 2014, pp. 476–488.

MR Endsley & DB Kaber. "Level of automation effects on performance, situation awareness and workload in a dynamic control task." *Ergonomics*, vol 42, 1999, pp. 462–492.

R. Thomson & JR Schoenherr. "Knowledge-to-Information Translation Training (KITT): An Adaptive Approach to Explainable Artificial Intelligence." In *International Conference on Human-Computer Interaction*, pp. 187–204. Springer, Cham, 2020.

R. Thomson. "The Cyber Domains: Understanding Expertise for Network Security," in *The Oxford Handbook of Expertise*. P.Ward and J. Gore Eds. Oxford Publishing. 2018.

SG Hart. "NASA-task load index (NASA-TLX); 20 years later." In *Proceedings of the human factors and ergonomics society annual meeting,* vol. 50, pp. 904–908. Sage CA: Los Angeles, CA: Sage publications, 2006.

SY Chien, Z. Semnani-Azad, M. Lewis, & K. Sycara. "Towards the development of an inter-cultural scale to measure trust in automation." In *International conference on cross-cultural design,* pp. 35–46. Springer, Cham, 2014.

TB Sheridan, & WL Verplank. "Human and computer control of undersea teleoperators." *Massachusetts Inst of Tech Cambridge Man-Machine Systems Lab*, 1978.