CPS: Driving Cyber-Physical Systems to Unsafe Operating Conditions by Timing DoS Attacks on Sensor Signals

Marina Krotofil
Hamburg University of Technology
Hamburg, Germany

Alvaro A. Cárdenas
University of Texas in Dallas
Richardson, TX 75080, USA

Bradley Manning
Autolive GmbH
Hamburg, Germany

Jason Larsen
IOActive, Inc.
Seattle, WA 98104, USA

ABSTRACT

DoS attacks on sensor measurements used for industrial control can cause the controller of the process to use *stale data*. If the DoS attack is not timed properly, the use of stale data by the controller will have limited impact on the process; however, if the attacker is able to launch the DoS attack at the correct time, the use of stale data can cause the controller to drive the system to an unsafe state.

Understanding the timing parameters of the physical processes does not only allow an attacker to construct a successful attack but also to maximize its impact (damage to the system). In this paper we use Tennessee Eastman challenge process to study an attacker that has to identify (in real-time) the optimal timing to launch a DoS attack. The choice of time to begin an attack is forward-looking, requiring the attacker to consider each opportunity against the possibility of a better opportunity in the future, and this lends itself to the theory of optimal stopping problems. In particular we study the applicability of the Best Choice Problem (also known as the Secretary Problem), quickest change detection, and statistical process outliers. Our analysis can be used to identify specific sensor measurements that need to be protected, and the time that security or safety teams required to respond to attacks, before they cause major damage.

Keywords

Cyber-physical systems, DoS attacks, optimal stopping problems, CUSUM, Tennessee Eastman process

1. INTRODUCTION

While compromising or disrupting devices or communication channels used to sense or control a physical system is a necessary requirement to attacks aimed at disrupting the physical process, the damage from the attack will be limited if the attacker is unable of manipulating the control system in a way to achieve her desired outcome in the physical world. After all, breaking into a system is not the same as breaking a system. In order to achieve a desired impact on a control system (like Stuxnet [14]), the attacker needs to assess how her attack will perform at the regulatory control level. Launching such an attack requires a different body of knowledge from the one used in IT security. In particular, attackers need to know how the physical process is controlled, and that includes knowledge of failure conditions of the equipment [15], control principles [30], knowledge of process behavior [19], and signal processing, etc.

In this paper we consider an attacker that can read a sensor signal for a given process variable, and then has to decide on a time to launch a DoS attack in order to “freeze” a certain process value above or below a set point in the controller’s memory. Most Programmable Logic Controllers (PLC) operate in a scan cycle architecture. During each cycle logic in the PLC uses the last saved input buffers obtained from sensor measurements to issue control command to the actuators. If input buffers are not updated because of the DoS attack, the last measurement received will be the one used by the PLC in each subsequent scan cycle. By doing so the attacker deceives the controller about the current state of the process and can cause compensating reactions which may bring the process into a state desired by the attacker, e.g. an unsafe state.

Typical sensor signals in process control either fluctuate around the set point or follow the dynamic changes in the process. In both cases the process variable exhibits a time series of low and high peaks. In order to move a sprocess into an unsafe state, the attacker should aim to freeze the sensor measurement at one of the optimal values of the process variable (low or high). However, because the attacker has to make the decision in real-time—where she needs to consider each opportunity against the possibility of a better opportunity in the future—the task the assaulter needs to address becomes a sequential decision problem where the attacker is presented with possible candidates at each time step, and has to make a decision immediately whether to launch an attack at this stage or not.

We formulate this challenge as an *optimal stopping time* problem for the attacker. In particular, we formulate the problem as a Best Choice Problem—also known as the Secretary Problem (SP)–in which the adversary is presented

ACSAC ’14, December 08 - 12 2014, New Orleans, LA, USA
Copyright 2014 ACM 978-1-4503-3005-3/14/12 ...$15.00.

with a time series of system states obtained by sensors and has to decide on the optimal time to attack based on current and past sensor measurements. In this paper we show that the problem the attacker has to solve is a non-trivial task in many practical situations as sensor measurements can be noisy and have sudden fluctuations. Using a plant-wide chemical control process simulation, we identify different types of sensor signals that can make the selection problem more challenging. We then explore different stopping rules and compare their effectiveness for selecting optimal sensor signal samples. Our results show that different stopping rules will be selected by different types of adversaries: risk-taking attackers (those who prefer to wait for a better opportunity at the risk of waiting for too long to launch an attack) and risk-averse adversaries (those who prefer to attack as soon as a reasonable attack opportunity emerges).

Our new threat assessment model can be used to measure the impact of the DoS attacks on sensor signals and to inform asset owners about the need to prioritize them for protection. In addition, our analysis can be used to identify the time interval (between the beginning of the DoS attack and the response from the security team) before the plant suffers either an emergency shutdown (if safety systems are in place) or a safety accident (if the attacker has disrupted safety systems as well).

2. TIMING AND CYBER-PHYSICAL SECURITY

In this paper we focus on Process Control Systems (PCS) which is a general term used to denote architectures, mechanisms and algorithms which enable processing of physical substances or manufacturing end products. Process industries include assembly lines, water treatment, pharmaceutical, food processing and other industries. In the past few decades plants have undergone tremendous modernization; technology became an enabler of efficiency but also a source of problems. What used to be a panel of relays is now an embedded computer, and what used to be a simple analog sensor is now an IP-enabled smart transmitter [24] with multiple wired and wireless communication modes, a large number of configuration possibilities, and even a web-server so that maintenance staff can calibrate and setup the device without approaching it.

Cyber-abuse in the IT domain do not generally depend on timing aspects. In certain instances such as during race conditions, Time-of-Check to Time-of-Use vulnerabilities, or cross-site scripting attacks that rely on getting access to session cookies before they expire, the attacker needs to make sure that their attack occurs within a tight window of time. In cyber-physical systems, however, timing takes an even more important role as the physical state of the system changes continuously over time, and during the system evolution in time, some states might be more vulnerable to attacks than others. Timing also characterizes the vulnerability of a system; e.g., it may take minutes for a chemical reactor to burst [31], hours to heat a tank of water or burn out a motor, and days to destroy centrifuges [14].Understanding the timing parameters of the physical processes does not only allow an attacker to construct a successful attack but also to maximize its impact.

The dynamic response of a processes variable changing from one state to another can be described with a simple model consisting of process gain, dead time, and time constant. The process gain describes how much the process will respond to a change in controller output, while the dead time and time constant describes how quickly the process will respond (Fig. 1). Precisely, dead time describes how long it takes before a process begins to respond to a change in controller output, and the time constant describes how fast the process responds once it has begun changing. Controlling the processes with large time constants is a challenging task causing operators' stress and fatigue [21]. The described timing parameters are not only important for the design of a control algorithm but also for the attacker to design an effective attack.

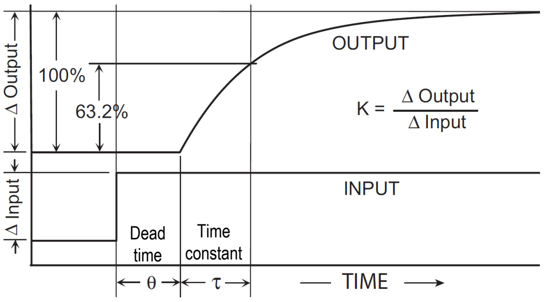


Figure 1: Time constants in process control, based on [30]

2.1 Adversary Model

The adversary's goal is to cause tangible impact on the process. In the physical domain, the attacker can either tamper with the sensor signals or modify the manipulated variables issued by the controller. In this work we limit our study to sensor signals. In particular we assume an attacker who aims to bring the process into an unsafe state by deceiving a controller about the current state of the process and thus forcing it to take harmful compensating actions. To do so the attacker can force the controller to believe that a process variable is below or above its set point. One way to achieve this is to forge the process variable by the means of a false-data injection attack. If the communication channel between a sensor and a controller has integrity protections (e.g., message authentication codes) and the attacker does not have key material, the attacker might opt to jam communication channel to prevent the controller from receiving process measurement updates. In the context of this paper we call this type of attack a DoS attack on a sensor signal.

As a rule, controllers store sensor signals in dedicated memory registers which are updated whenever a new value arrives. During the DoS attack, the input register assigned to storing measurements of a particular sensor will not be overwritten by fresh values. Therefore, the last process value which reached the controller before the attack, will be used by the PLC for the duration of the DoS attack. As a result, the controller will generate control commands based on the last received reading. A DoS attack when compared to a false-data injection or integrity attack is different in that the adversary does not have direct influence on the "attack value". Instead an adversary can take advantage of the timing parameters of an attack, such as the starting time t_a of the attack and the duration T_a.

The impact of the attacks on PCS are sensitive to the specific state of the system, in particular, attacks might only be effective if the process variable is above (or below) a certain threshold. The higher the attack process variable is chosen beyond the threshold, the higher the impact. Moreover, since DoS attacks are easy to detect, one of the goals of the attacker is to achieve its disruption objective as soon as possible after the attack is launched. Therefore, the attacker should aim at launching a DoS attack at the time the process variable of interest reaches its local maximum or minimum (a more vulnerable state).

The attacker faces the following problem: given a time-series that exhibits a sequence of peaks and valleys of different amplitude, she has to select one of the peaks to launch a DoS attack in real time. If the attacker strikes too soon, she might lose the opportunity of having a higher impact on the system if she had waited longer (i.e., if the process variable would reach higher value later in the future). However, if the attacker waits for too long, the process variable might not reach a more vulnerable state than previously observed, and the adversary might miss the opportunity to cause maximal damage, or even have the implanted attack tools detected before they have the chance to launch the attack.

The problem of selecting a good time to attack can be framed as an *optimal stopping problem*. These class of problems are concerned with the challenge of choosing the time to take a particular action based on sequentially observed random variables in order to maximize an expected payoff. Such class of optimal stopping decision tasks in which the binary decision to either stop or continue the search depends only on relative ranks is modeled as Best Choice Problem, also known as the Secretary Problem [9].

2.2 Susceptibility of Control Equipment to Stale Data Attack

Almost every device in process control equipment needs to deal with stale data but it is rarely documented in open literature. In many environments the bandwidth required to report the current state of every point (field devices) in the system exceeds the capacity of the communications channel. This is particularly prevalent when the data is communicated over long-haul serial communications. Open-source and proprietary protocols both describe themselves as "report by exception" meaning the data is reported to the server only when it has changed significantly. In many cases the amount of change required for the process value to be reported is configurable but the exact rules may still be opaque.

Extensive testing of Industrial Control Systems was performed at the Idaho National Labs and other facilities [6]. During those tests a number of scenarios were found where stale data could be maintained in a system for indefinite periods of time. No general technique is known that is applicable to every device so we describe some of the implementation details that have previously allowed for stale data.

The most common root cause of stale data is when the protocol stack was implemented in a stateless manner. The master maintains a set of timers for the data tied to individual data points or a set of data points assigned to a class. When data is received by exception the timer is restarted. If the data changes often enough, the master never sends a poll request. If the timer expires, the point is placed into a point list to be requested during the next poll. If many cases, the timer is reset when the poll request is sent with the assumption that the point value will be returned in the next reply. In other cases, the timer is only reset upon a successful receipt of the updated data.

The most common finding was if the TCP session was simply maintained without transferring any data, the sensor was never reported as offline. The second most common finding was that the protocol lacked replay protection and old messages could simply be resent. Also more complex attacks have been shown possible. Thus, in many implementations the server listens on both the UDP and TCP ports. Messages received via TCP, UDP, or serial communication were routed to the same processing code without checking which communications mechanism sent the message. An exception report could be sent via the UDP port but it would be merged with the ones received via an existing TCP session. In case the sequence numbers are used to detect missing and retransmitted data, empty acknowledgments with higher sequence numbers could be sent via UDP to increment the internal sequence number of the controller. When a legitimate message arrived at the TCP port, it would be discarded as retransmissions.

Furthermore, a number of examples can be formed where interfering with a data channel is possible but fully viewing and manipulating the data channel is not. The most common case is networking equipment where the attacker only has access to the administrative interface but not to the data plane. For example, in common routers there is a "fast path" where packets are routed in custom designed application-specific integrated circuits (ASIC) and the "slow path" where packets are transferred to the CPU. Access to the administrative interface can give code execution on the device, but only packets transiting the slow path can be observed and manipulated. If all packets are requested by the slow path, the CPU is saturated resulting in a DOS on the communications link. If packets are sampled periodically, they can be examined and with that allowing an attacker to perform a DoS attack at the opportune time.

3. BEST CHOICE AND OPTIMAL STOPPING PROBLEMS

In this section we introduce the theoretical background used in our formulation.

3.1 Best Choice Problem

In the standard version of the Best Choice Problem, a finite and known number n of items (or alternatives) are presented to a decision maker (DM) sequentially, one a time, in a random order. Time is discrete. At any period the DM is able to rank all the items that have been observed in terms of their desirability or quality. For each item inspected the DM must either accept it, in which case the search process is terminated, or reject it and then the next item in the random order is presented and the DM faces the same problem as before. The DM's objective is to maximize the probability of selecting the best item of all the n items available.

The classical Best Choice Problem is formulated in terms of a hiring manager identifying the best secretary (the Secretary Problem) it has interviewed among all applicants. This problem is formulated in terms of six assumptions:

1. There is only one position available.
2. The number of applicants, n, is finite and known to the DM.
3. The n applicants are interviewed sequentially, one at a time, in a random order; consequently, each of the $n!$ orders is equally likely.
4. The DM can rank all the n applicants from best to worst without ties. The decision to either accept or

reject an applicant in a given period is based only on the relative ranks of those applicants interviewed so far.

5. Once rejected, an applicant cannot later be recalled.

6. The DM is satisfied with nothing but the best. (Her payoff is 1, if she chooses the best of the n applicants, and 0 otherwise.)

Note that an applicant is accepted only if it is relatively best among those already observed. A relatively best applicant is called a *candidate*. The stopping rule suggested by the Best Choice Problem theory is the following: do not make any offer to the first n/e candidates (where e is the base of the natural logarithm) and after that, make an offer to the first candidate whose value exceeds the value of all candidates seen so far (or proceed to the last applicant if this never occurs, such case is called non-selection). In other words, the algorithm starts with a *learning phase* in which the DM sees n/e candidates and sets an *aspiration level* equal to the highest value seen in the learning phase. After that, the DM hires the first candidate that exceeds the aspiration level. The main result of the best choice problem states that the optimal stopping rule can select the best candidate with at least $(1/e)$ probability.

It has been recognized that the classical assumptions place more constraints on the observation and selection than would generally apply in practice [12]. Relaxing one or more assumptions for a more realistic formulation of the standard assumptions has attracted attention in the research community. In this paper we consider the classical solution, and a recent result that assumes the order in which the candidates arrive is not completely random, but has a probability distribution satisfying a hazard rate condition [20]. This assumption is commonly used in standard engineering applications and states that given that the value of a candidate is not less than y, the likelihood that it is equal to y increases as y increases. As an example, Gaussian, uniform, and exponential distributions satisfy this property. Under these assumptions it was shown that the learning period falls from n/e to $n/log(n)$, meaning that it is enough to observe a much smaller number of candidates to set the optimal aspiration level and have a similar probability of success. Since the probability of detecting an intrusion increases with time, having a shorter learning phase is beneficial to the attacker.

Fig. 2 illustrates formulation of the Secretary Problem and its solution in the context of an arbitrary sensor signal. We call learning phase as an *observation window* and we refer to the selection phase as an *attack window*. Notice that in this case the maximum selected is lower than the overall maximum in the attack window.

3.2 Peak Detection

In the time series of the physical phenomenon, each time sample X_i is heavily correlated with the next sample X_{i+1}. Thus, if a process variable (e.g. temperature) is increasing, it cannot drop radically in the next time instance. As follows from the SP solution, upon completion of the learning phase the attacker should select the first sample, whose value exceeds aspiration level. By doing so the attacker can miss the opportunity to select an even higher value as in case of an upwards trend where the process measurement will keep increasing until it reaches its local peak. Therefore, in contrast the static choice rule, the attacker may incorporate expectations about the future into her decision process.

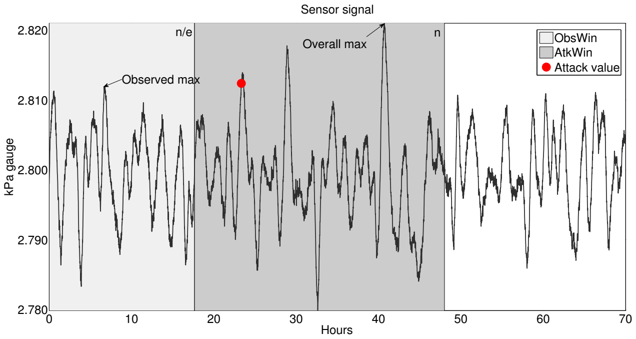


Figure 2: Illustration of the Secretary Problem solution applied to a sensor signal

The problem of identifying a signal peak is exacerbated by the fact that process variables are *noisy* and therefore an upward trend might be followed by a quick drop, followed again by an even higher gain. We propose two algorithms for peak detection.

3.2.1 *Forward Looking Search*

In our initial approach we add a low-pass filter to the signal to smooth out short-term fluctuations and highlight longer-term trends. In this case the choice between stopping or continuing to search at sample X_i is determined by the difference between the stopping value and the continuation value X_{i+1}. This allows a peak to be identified as soon as a downward trend in a smoothed signal is detected (e.g., three consecutive measurement drops). The optimal smoothing interval depends on the sampling frequency of the signal.

3.2.2 *Change Detection*

A more sound approach to peak detection is quickest change detection theory. Change detection tests are statistical techniques that allow identification of a possible drift, abrupt, and and sudden changes in data series at an unknown time. Cumulative sum (CUSUM) is one of the most commonly used algorithms for change detection problems. Most optimal stopping rules are parametric and require a priory knowledge about the statistics of the signal and the size of the expected shift. However, in several practical cases the underlying distribution and the magnitude of shifts are both unknown. In such cases the non-parametric scheme (NCUSUM) is a more suitable choice. Let the sensor time series be represented by a series X_i. Our goal is to identify the peak of this time series as soon as possible. To do this we define a new time series S_i. In particular we use two NCUSUM statistics for each signal, S_i^+ to detect a local maximum in the signal, and S_i^- to detect a local minimum. The NCUSUM is initialized with $S_0^+ = 0$ and $S_0^- = 0$ and updates as follows:

$$S_i^+ = \max(0, |X_{i-1} - X_i| + S_{i-1}^+),$$
$$S_i^- = \max(0, |X_i - X_{i-1}| + S_{i-1}^-),$$

An alarm is triggered whenever $S_i > h$, where h is a threshold that can be used as a parameter to control the tradeoff between the rate of false alarms and the delay for detecting the local maximum or minimum of the signal. Although NCUSUM statistic is general and can be applied to any signal, the threshold h has to be learned for specific types of sensor signals.

3.3 Heuristic Algorithm

Generally speaking, the SP allows an attacker to select the best possible "outlier" sample (due to noise) in the time series. Although SP is based on a sound mathematical theory, one of the disadvantages of the SP optimal solution is the high number of non-selections (the last sample in the time series is taken because the attacker ran out of time). To avoid non-selections we propose an alternative heuristic solution which we call the Outlier Test (OT).

In process control it is not uncommon to assume process measurement noise follows a Gaussian distribution, which is characterized by the mean value μ and standard deviation σ. These parameters can be estimated for each signal (during the observation period), and then, during the attack window, the attacker can use this knowledge to detect outliers. According to the "three-sigma rule" in a normal distribution nearly all values (0,9973%) lie within three standard deviations of the mean. In this case, the attacker will select the first sample in the attack window such that $X_i > \mu + 3\sigma$ or $X_i < \mu - 3\sigma$. Alternatively these thresholds can be lowered to values that may be achieved faster if the attacker has a time deadline. A standard score curve can be used to determine the optimal OT threshold h.

4. SIMULATION SETUP

One of the challenges of cyber-physical security research is the lack of large-scale test beds to allow the study of complex attacks and their effects on physical processes. To mitigate this problem researches can leverage simulation models of realistic industrial plants [7, 2, 5] which have been developed for the process control community to focus on issues important to the industry and to allow comparison of research results. These models can be adapted to study the various aspects of cyber-physical exploitation. In this work we use the full plant-wide control problem proposed by Down and Vogel [7].

4.1 Tennessee Eastman Challenge Process

The Tennessee Eastman (TE) process is a modified model of a real plant-wide industrial process. The authors intentionally omitted certain specific details of the process to protect its proprietary nature. This makes TE problem an excellent case study because the *a priory* information about process is limited and thus allows emulation of the "grey-box" exploitation use cases. For our empirical analysis we use the TE Matlab model developed by Ricker [27]. The plant produces two liquid (*liq*) products from four fresh gaseous (g) reactants involving two irreversible exothermic reactions composed of chemicals A, C, D, E, G and H:

$$\begin{array}{ll} A(g) + C(g) + D(g) \to G(liq), & \text{Product 1,} \\ A(g) + C(g) + E(g) \to H(liq), & \text{Product 2.} \end{array}$$

The process has five major operation units: the reactor, the product condenser, a vapor-liquid separator, a recycle compressor and a product stripper as shown in Fig. 3. The gaseous reactants and products are not specifically identified. Feed C is not pure and consists of 48.5% A and 51% C. The byproducts and inert gases are purged from the system in the vapor phase using a vapor-liquid separator whereas products G and H exit the stripper base and are separated in a downstream refining section. The plant has 12 valves for manipulation, and in total 41 measurements (with added measurement noise) are involved in process monitoring. The first 22 measurements are continuous, the rest are sampled composition analysis from chromatographs with a delay of 0.1 or 0.25 hours, depending on the process variable. The simulation model control scheme consists of 18 proportional-integral (PI) controllers, 16 process measurements XMEAS{1;2;3;4;5;7;8;9;10;11;12;14;15;17;31;40} and 9 set points which form 8 multivariable control loops and 1 single feedback control loop [17]. The full notation and units of process characteristics including operation constraints can be found in [7]. The default simulation time of a single experiment is 72 hours.

The process description includes a flowsheet, steady-state material and energy balance as well as the operational constraints of both the optimal steady-state operations and the process shutdown limits. Depending on her goal (economic or physical disruption), the attacker would have to violate these specified constraints. In addition, the original problem statement includes typical set points and load changes, which along with other listed disturbances illustrate different aspects of process operations. In total there are 20 disturbances modes IDV{1-20} and four set point changes. This information is valuable to the attacker as any change in operations causes variations in the process behavior which in turn is visible in the process measurements (sensor signals).

Depending on the noise level and shape of the signal, sensor signals in TE process can be roughly divided into 4 distinct groups (Fig. 4). Type 1 is characterized by a few distinct peaks and a low noise level. Type 2 is distinguished by the multiple noisy signal peaks. Type 3 can be described as a very noisy variation of Type 1 signal. The type 4 signal distinguishes itself by the overall slow signal amplitude change with high amplitude noise. Depending on the state of process, sensor signals can change their properties substantially. For example, A feed flow F_A is of Type 4 in a steady state, of Type 2 under disturbance IDV(11) and of Type 1 under IDV(8).

Initial model does not allow any randomness in the simulations to guarantee the repeatability of the plant operation disturbances. It means that each simulation run produced identical results. In order to obtain statistically significant results we modified the original code by generating a new seed for the random number generator for each run while preserving underlying dynamics of process behavior.

4.2 DoS Attack Model

Let $X_i(t)$ be a measurement of sensor i at time t, where $0 \leq t \leq T$, and T the duration of the simulation; time is discrete. The attack interval T_a is arbitrary and is limited to the simulation run time. In our setting, we simulate manipulated sensor readings X_i' as follows:

$$X_i'(t) = \begin{cases} X_i(t), & \text{for } t \notin T_a \\ X_i^a(t), & \text{for } t \in T_a, \end{cases}$$

where $X_i^a(t)$ is the modified reading (attack value).

During a *DoS attack*, new sensor measurements do not reach the controller. If the attack starts at time t_a, we have:

$$X_i^a(t) = X_i(t_a - 1).$$

where X_t^a is the stale data reading (the last value received by the controller before the DoS attack).

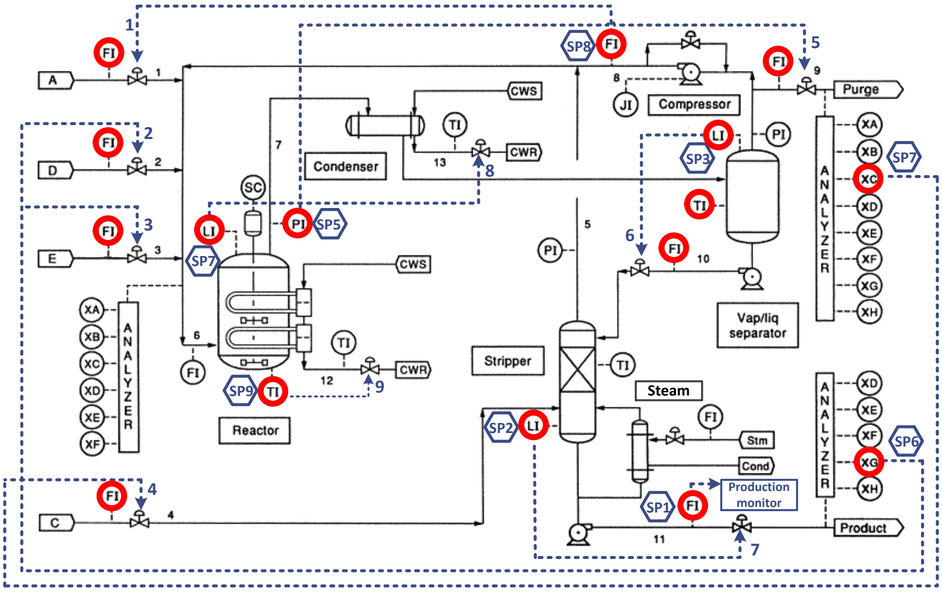
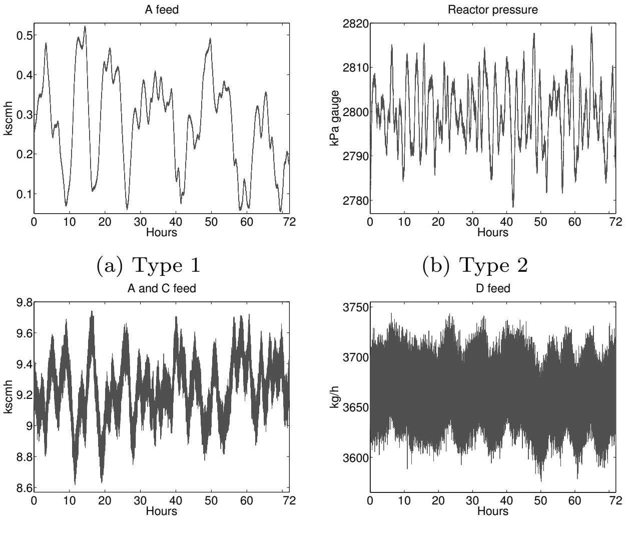


Figure 3: Plantwide chemical process under control–based on [17]



(a) Type 1

(b) Type 2

(c) Type 3

(d) Type 4

Figure 4: Different types of sensor signals in TE process

5. EXPERIMENTAL RESULTS

The TE model execution starts with the predefined base values. The "warm-up" phase of the plant lasts for about two hours and is excluded from our analysis. In the original implementation the process data is downsampled to 100 samples per hour (s/h) before being stored in the Matlab workspace. We conduct our analysis based on the sampling rate of f_s=2000 s/h which is used during the actual real time simulations. We omitted analysis of XMEAS{31;41} as these measurements are not real time. Unless specified differently, the experimental results are conducted with IDV(8) active which stands for the variation in the reactor feed. This disturbance is successfully absorbed by the process and does not impact operations but causes noticeable unpredictable deviations in sensor signals. All statistical results are averaged based on 50 simulation runs. The 95% confidence interval is calculated using Student's t-distribution but due to space limitations are not included into paper.

5.1 Evaluation Metric: Shortest Shutdown Time

In our scenario, we assume an attacker whose goal is to force the physical process into unsafe state and cause its shutdown. To evaluate the result of such an attack we select the Shutdown Time (SDT) as a metric that measures the time that the process is able to maintain safe operating conditions (e.g., maintain the pressure in the reactor tank below $3,000kPa$) after the attack is started. A longer SDT is unfavorable to the attacker as plant operators have more time to take corrective measures to bring the process back into a safe state. In order to be able to compare the results of the proposed approaches, we first obtain reference values of the shortest possible safety time for the attack on each individual sensor. The reference values are selected by looking at the overall lowest (min) and highest (max) process values in each individual simulation run. These attacks would be infeasible in real-time because they would require an attacker to go back in time to select the best value. We refer to these SDTs as *optimum* (Tbl. 1).

As can be seen, the sensitivity of control loops to deception attacks varies greatly with a SDT range from minutes to more than 20 hours. We did not include results on XMEAS{10;11} as no attack on these sensors places system into an unsafe state. Also, only five F_A^{min} attack instances triggered a process shutdown. This is because susceptibility of the process to the attack on A feed depends not only on the attack value but also on the overall balance of A- and C-components. At the same time, attacks P_{reac}^{max} and F_A^{max} do not drive process into an unsafe state. This means an assaulter planning an attack on e.g. reactor pressure should strike only at the minimum peaks.

5.1.1 Without Peak Detection

In this subsection we analyze the attacker's prospects of selecting the highest possible process value in real time ap-

XMEAS		Variable name	Optimum	Secretary, n/e			Secretary, n/log(n)			Outlier Test, $\sigma = 2.0$		
			SDT,h	SDT,h	Error,%	NS,%	SDT,h	Error,%	NS,%	SDT,h	Error,%	NS,%
(1)	min	A-feed	22.22	-	13.96	40	-	42.85	12	-	18.74	70
	max	rate										
(2)	min	D-feed	5.15	5.93	8.52	26	6.05	19.57	0	6.72	42.93	0
	max	rate	3.69	3.85	6.38	20	4.27	13.64	4	6.36	42.15	0
(3)	min	E-feed	4.29	4.46	8.13	38	4.55	17.65	2	5.26	44.87	0
	max	rate	2.83	3.51	7.82	36	3.32	19.43	2	4.44	45.18	0
(4)	min	C-feed	1.05	1.78	12.44	34	1.99	35.80	4	2.07	29.15	12
	max	rate	1.78	2.38	18.83	44	1.93	34.45	8	2.20	28.44	6
(5)	min	Recycle	4.39	5.44	17.12	32	8.73	39.17	2	6.77	28.92	14
	max	flow	9.17	9.90	25.12	40	10.11	42.66	0	10.17	35.44	12
(7)	min	Reactor	8.56	8.2	23.41	34	8.21	39.20	8	8.51	25.93	8
	max	pressure										
(8)	min	Reactor	2.37	2.90	14.57	34	3.31	30.05	2	3.96	44.98	0
	max	level	2.73	3.08	11.13	32	3.40	24.20	0	3.71	37.69	0
(9)	min	Reactor	1.34	1.39	4.06	30	1.45	13.54	8	1.75	45.25	0
	max	temper.	0.65	0.69	3.74	40	0.70	13.56	6	3.71	37.69	0
(12)	min	Separator	5.50	6.92	17.10	36	8.74	40.67	0	8.15	31.18	0
	max	level	3.49	4.28	15.97	30	5.86	41.60	2	5.96	36.10	4
(14)	min	Separator	12.03	12.10	17.50	34	11.88	39.16	0	12.58	34.01	0
	max	underflow	11.58	11.13	20.22	26	11.69	40.70	2	12.15	34.33	2
(15)	min	Stripper	6.39	7.96	20.28	20	11.06	49.43	4	9.70	35.63	14
	max	level	6.35	8.54	23.76	34	13.05	56.36	2	9.71	34.01	10
(17)	min	Stripper	1.86	2.33	4.89	18	3.37	13.91	4	6.75	42.77	0
	max	underflow	1.33	2.36	7.07	24	3.10	19.99	2	7.14	43.30	0

Table 1: Simulation results of the approaches to the Best Choice Problem solution

plying SP approach, without dealing with the fact that sensor measurements are correlated. To begin, the attacker has to decide on two parameters in the context of the secretary problem, namely on the number of samples or alternatives n she is going to consider, and the duration of the learning phase. For simplicity we measure n in hours. In this case, with a time frame of 48 hours, the number of alternatives is equal to $48 \times f_s$. We test both lengths of the learning window n/e and $n/log(n)$. There is no specific requirement to the size of learning window for the Outlier Test. We keep it set to n/e to obtain results comparable with the "classic" secretary approach. Additionally, the Outlier Test requires a decision on the detection threshold. We set it to $\sigma = 2$ which accounts for 95.45% of the data set. We evaluate the approaches based on two metrics: (1) error in selecting the max value in the attack window, in %; (2) number of non-selections (NS) – when no sample is higher than the one observed in the observation window. To evaluate the influence of the error in selecting the highest value we also measure the SDT for each attack instance. The results of the simulations are summarized in Table 1.

Although the classic SP solution with the size of learning window n/e results in the lowest error when selecting the best sample, its shortcoming is a high number of non-selections. This side effect is disadvantageous for the attacker as she either has to choose a clearly suboptimal candidate (last sample in the attack window) or decide to not launch an attack. Cutting the learning window to $n/log(n)$ or applying the Outlier Test substantially reduces the number of non-selections albeit at the cost of selecting a lower attack value and an increased SDT. As the table indicates, most of the control loops are sensitive to the magnitude of the selected attack value.

Another feature of the classic SP solution is at odds with the attacker requirements in that it prescribes a long learning phase before selecting a candidate. The length of the observation window together with a time to selection (TTS) constitute a time to attack (TTA). The longer TTA, the greater chance for the attacker to be detected (also accidentally). Table 2 presents a comparison between the approaches based on the aggregated (averaged) results from Tbl. 1 and results on TTA and TTS. Although OT and SP with $n/log(n)$ algorithms demonstrate roughly similar results in the accuracy of selecting the maximum value and time before making a selection, the latter approach benefits from a drastically smaller time to attack with a very small observation window. It is therefore is more advantageous to the attacker.

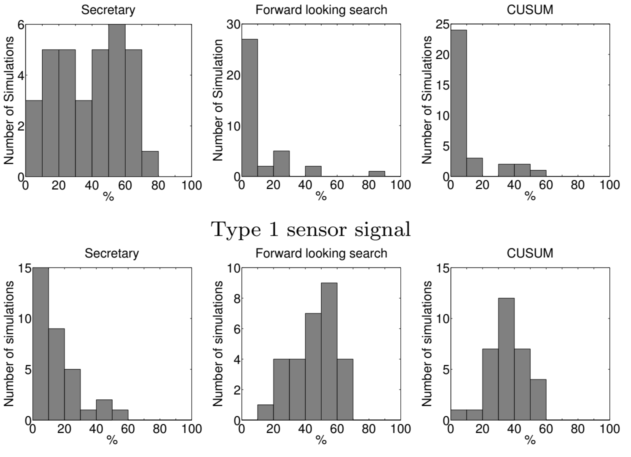
	Error	TTS	TTA	NS
SP, n/e	13.78 %	11.52 h	27.17 h	37.77 %
SP, $n/log(n)$	31.26 %	7.63 h	9.61 h	3.36%
OT, $\sigma = 2.0$	36.93 %	5.32 h	20.98 h	3.9 %

Table 2: Aggregated comparison between the approaches

5.1.2 Peak Detection

To take advantage that sensor signals are correlated, the attacker may add a peak detection step into her attack strategy in order to wait if the signal shows a trend. In the forward looking search approach we apply a simple moving average with a smoothing interval $m = 250$ to filter out high frequency noise.

The threshold in the CUSUM algorithm depends on the scale of the change an assaulter would like to detect. Thus, for the noisy signals the threshold should be selected higher than for the low noise signals. We determined optimal thresholds being $h = 0.005$ for signal or Type 1 and 2, and $h =$



Type 1 sensor signal

Type 3 sensor signal

Figure 5: Distribution of the error in selecting highest possible sample, without and with peak detection

0.015 for signal of Type 3. There are no distinct peaks in the signals of Type 4. We compare the peak detection approaches with the classic SP solution as it delivers the most optimal results in selecting the max value (without peak detection). We illustrate the comparison of the results in the form of a histogram which represents the distribution of the error in selecting an optimal sample (Fig. 5).

As can be seen, peak detection approaches deliver superior results in comparison to the SP algorithm for the sensor signal of Type 1 (first row in the figure). This is because in the SP search finishes too early by ignoring the correlations of the signal, thus yielding high error in selecting the max value. At the same time the low noise level allows accurate detection of the signal peak. In contrast, the proposed approaches do not perform well in the case of noisy signal. Thus, noise corrupts the CUSUM statistics resulting in high false alarm rate. Although smoothing reduces the noise in a signal, it introduces a delay of smoothed signal by $(m-1)/2$ samples resulting in the peak detection when the actual signal is already decaying. Vice versa, the SP delivers a lower error while dealing with noisy signals as it sets up the aspiration value based on the highest outlier value in the observation window.

Whereas enhancing SP solution with peak detection may in certain cases substantially increase the attacker's chances in selecting max value, any peak detection algorithm requires to be tuned first to match sensor signal properties.

5.1.3 Attacks in a Steady State

So far we have reported results with disturbance IDV (8) active. However, one of the primary goals of process control is to keep the process as close as possible to its optimal steady state (without disturbances). In the TE process variables in a steady state do not deviate much from the set points and accurately follow the Gaussian distribution. Most of the sensor signals are of Type 4 (with few signals of Type 3). Results from Table 3 shows that relative to each other, all approaches demonstrate performance similar to the use case from 5.1.1 with the exception of the NS statistics for OT. Even with a higher threshold of $\sigma = 3.5$ the Outlier Test delivers diminishingly small number of non-selections. Also, in a steady state the observation window for the OT can be substantially reduced. For most of the sensor signals it is sufficient to observe only 5 hours of samples in order accurately identify μ and σ. With that the TTA in the Outlier Test becomes comparable to the one of the secretary approach with reduced learning window $n/log(n)$.

	Error	TTS	TTA	NS
SP, n/e	3.20 %	2.87 h	28.53 h	37.09 %
SP, $n/log(n))$	9.09 %	8.39 h	10.37 h	8.64 %
OT, $\sigma = 3.5$	7.82 %	6.39 h	22.02 h	1.5 %

Table 3: Comparison between the approaches, steady state

Due to the minimal variations in the process measurements all approaches demonstrate a lower error in selecting best possible value. However, for the same reason the selected attack values are also lower. Moreover, in a steady state the process is more resilient to the attacks on certain individual control loops. For instance, it becomes impossible to bring the process into unsafe state with any attack on XMEAS(1) and the SDT for the attacks on XMEAS{5;12;17} increases more than twice. Therefore, it might not be rewarding for the attacker to strike at a steady state.

5.2 Discussion and Future Areas of Research

After obtaining access to the process measurements the attacker faces a number of uncertainties. She neither knows the process variable range nor the sensitivity of the process to the magnitude of the manipulations. What is more important, the adversary has no knowledge about the time constants of individual control loops and specifics of disturbances propagation. It means that the attacker is not certain which attack value to choose and for how long to carry out her assault. The latter is crucial knowledge, for instance for planning concealing activities. Hence, to maximize the impact and minimize attack duration the attacker should try to select the highest or lowest process value possible.

The results of our study shows that the characteristics of the sensor signals even in the same facility are very dissimilar and the attack strategies deliver radically different results when applied to the different types of process measurements at the different plant states and operating modes. Attacking without knowledge about current state of the plant is highly likely to only result in nuisance rather than an actual disruption. In contrast, a knowledgeable attacker may bring the system down in a matter of minutes leaving operators no chance to respond with countermeasures (e.g. [31]). Therefore destructive capabilities in the cyber-physical domain predominantly exist in relation to a specific target and knowledge on process dynamics (such as results from Table 1) are only valid for a specific process control scheme. We would have to conduct a separate analysis for the other control schemes of the TE process, such as [22, 28, 19].

The attacker can do her home work well and design part of the attack in advance; however she will have to tune the attack locally through reconnaissance activities such as changing configuration parameters, manipulating process variables or turning components on and off while observing the system's reaction. From the defense point of view, such short-term “testing” process deviations can be detected by the process-aware anomaly detection solutions [23, 4]. Below we describe further considerations and areas of future research.

5.2.1 Impact of the Sampling Frequency

The sampling rate of the sensor signals changes its noise

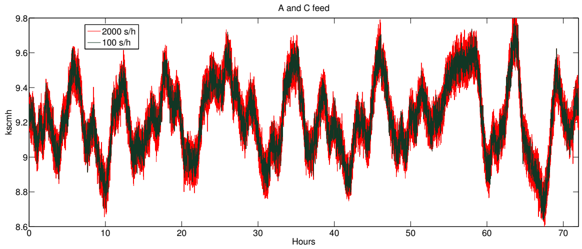


Figure 6: Impact of the sampling frequency

profile. Therefore we also investigated the impact of the f_s. As anticipated, the lower signal sampling rate results in lower error in selecting the best sample due to reduced noise level. However, the selected attack value is also lower if compared to the attack on the signal sampled at a higher rate due to the reduced number of the high amplitude outlier samples (Fig. 6). Depending on the control loop sensitivity, we could observe increase in shutdown time up to 3 hours.

5.2.2 Detection of Plant State Change

Reference value learned in the observation window is only valid for a particular plant state. In practice, the process periodically undergoes through the periods of changes in its operating conditions such as updates of set points, operating modes, production loads, disturbances, etc. The attacker needs to be able to detect such changes quickly in order to adapt her attack strategy to new circumstances. Fig. 7 demonstrates detection of the A/C feed ratio change using CUSUM algorithm. With a threshold $h = 0.001$ such change can be detected in 8.5 min and with $h = 0.005$ in 12.5 min. Once the change is detected, the attacker can either reset her learning phase straight away or wait for some time and see whether process state will keep changing.

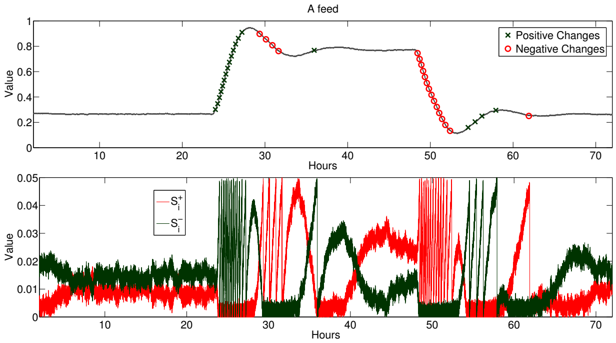
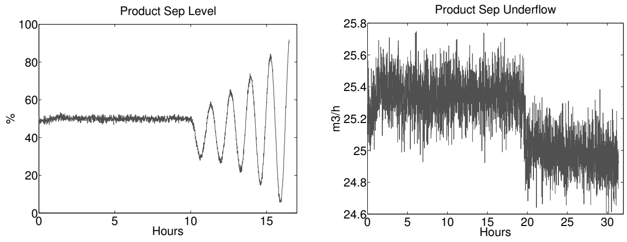


Figure 7: Detection of the plant state change, $h = 0.005$

5.2.3 Chaining Attacks

As was mentioned in 5.1.3, even after executing a successful DoS attack in a steady state it would take a long time to bring the process into unsafe state. In order to achieve shorter SDT, the attacker would need to disturb the process first to cause greater deviation of the process measurements (so that she could select a higher or lower attack value). Causing a plant-wide disturbance might be hard, however the attacker can "chain" two DoS attacks to accomplish her goal. For instance, F_{sep}^{min} attack on a separator underflow causes an oscillation effect on the separator level (Fig. 8a). After 30 min the separator level L_{sep} reaches 30%. The attacker can use CUSUM algorithm to detect the change and launch a DoS attack on L_{sep} when its value reaches its lowest point. In this way the shutdown can be reached in 3.43 hours in comparison to 12.03 hours if the assaulter would execute a direct attack on separator level sensor in a steady state. Similarly, L_{sep}^{min} attack causes an immediate step change of the separator underflow (Fig. 8a) which can be quickly and accurately detected and used for the successive DoS attack.



(a) Oscillation effect of F_{sep}^{min} attack

(b) Step change effect of L_{sep}^{min} attack

Figure 8: Examples of chaining attacks

6. RELATED WORK

Securing process control infrastructure and control communication is the first step to safe and secure operations. A large body of literature on Industrial Control Systems (ICS) security looks at intercepting and manipulating the traffic [29], infection by malware [3], access by unauthorized users [18] and addresses the threats by designing ICS-specific defenses such as intrusion detection systems [32, 13], authentication and encryption schemes [8], access control [26], code verification [25] and others.

Physical processes and their particular states are inherently time dependent. The important role of the timing parameters in cyber-physical security was already demonstrated in few academic works. Thus it was shown that process-aware segmentation of the control network increases survivability of the process and extends its time to shutdown [10]. In another work authors studied the effectiveness of remotely executed cyber attacks on a valve in a Boiling Water Power Plant [11]. PLC task scheduling turned to have one of the major impacts on the attack outcome. In [1] authors discuss synchronization as a timing parameter crucial for the stability of the power grid and provide an understanding of the impact of timing uncertainty on the system model accuracy needed to achieve timely situational intelligence. A heuristic triangle approximation algorithm from [16] can be used for peak detection in sensor signals.

7. CONCLUSION

In this paper we introduced the problem of timing DoS attacks based on real-time measurement of process values. We used the TE process to illustrate our approach, but our basic methodology is applicable to any cyber-physical system.

The results of our study shows that the characteristics of the sensor signals even in the same facility are very dissimilar and the attack strategies deliver radically different results when applied to the different types of process measurements. We also showed that applying peak detection algorithms for dealing with correlated time series can potentially improve the performance of the attacker. However noisy signals can render peak detection ineffective.

In general, it is not possible to give definite conclusions regarding which of the proposed approaches is more effective. All approaches have their own advantages and disadvantages, and their performance largely depends on the type of the signal under analysis and the state of the plant. For instance, NCUSUM can detect changes to the plant state (e.g., in case of the reference variable change) as it is quick and can be easily tuned; however, it was not a significant addition to the problem of detecting peaks in noisy signals.

Overall, the classic secretary approach delivers the best results at the cost of having a long learning phase and a relatively high number of non-selections; however, it consistently delivers good results regardless of the plant state and signal shape. This is particularly useful in the context of black-box exploitation, when the attacker has no a prior knowledge about sensor signals properties. Adversaries that do not want to take the risk of ending the observation period without making a decision may select the SP solution with the reduced learning window $n/log(n)$ or the Outlier Test in order to shorten the time to attack.

In future work we plan studying how attacking multiple sensor signals can affect the system, and to include DoS attacks on control signals.

Acknowledgments

The work of Alvaro A. Cardenas was supported in part by award 70NANB14H236 from U.S. Department of Commerce, National Institute of Standards and Technology.

8. REFERENCES

[1] J. Amelot, D. Anand, T. Nelson, G. Stenbakken, Y.-S. Li-Baboud, and J. Moyne. Towards Timely Intelligence in the Power Grid. In *44th Annual PTTI Meeting*, 2012.

[2] R. Bell and K. Åström. *Dynamic Models for Boiler-Turbine aAlternator Units: Data Logs and Parameter Estimation for a 160 MW Unit.* 1987.

[3] A. Carcano, I. N. Fovino, M. Masera, and A. Trombetta. SCADA malware, a proof of concept. In *CRITIS'08*, pages 211–222, 2008.

[4] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry. Attacks against process control systems: risk assessment, detection, and response. In *ASIACCS'11*, pages 355–366, 2011.

[5] R. Chen, K. Dave, T. J. McAvoy, and M. Luyben. A Nonlinear Dynamic Model of a Vinyl Acetate Process. *Industrial & Engineering Chemistry Research*, 42(20):4478–4487, 2003.

[6] Control Systems Security Program. Common Cybersecurity Vulnerabilities in Industrial Control Systems. 2011.

[7] J. J. Downs and E. F. Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255, 1993.

[8] I. Fovino, A. Carcano, M. Masera, and A. Trombetta. Design and implementation of a secure modbus protocol. *CIP III*, pages 83–96, 2009.

[9] P. Freeman. The secretary problem and its extensions: A review. *International Statistical Review/Revue Internationale de Statistique*, pages 189–206, 1983.

[10] B. Genge and C. Siaterlis. An Experimental Study on the Impact of Network Segmentation to the Resilience of Physical Processes. In *LNCS*, volume 7289, pages 121–134. 2012.

[11] B. Genge, C. Siaterlis, and M. Hohenadell. Impact of network infrastructure parameters to the effectiveness of cyber attacks against industrial control systems. *IJCCC*, 7(4):673–686, 2012.

[12] J. Gilbert and F. Mosteller. Recognizing the Maximum of a Sequence. *Journal of the American Statistical Assosiation*, pages 35–73, 1966.

[13] D. Hadžiosmanović, D. Bolzoni, and P. Hartel. A log mining approach for process monitoring in SCADA. *IJIS*, 11(4):231–251, 2012.

[14] R. Langner. To kill a centrifuge. Technical report, Langner Communications, 2013.

[15] J. Larsen. Breakage. *Black Hat Federal*, 2008.

[16] J. Larsen. Miniaturization. *Black Hat USA*, 2014.

[17] T. Larsson, K. Hestetun, E. Hovland, and S. Skogestad. Self-optimizing control of a large-scale plant: The Tennessee Eastmann process. *Ind. Eng. Chem. Res.*, 40(22):4488–4901, 2001.

[18] E. Leverett and R. Wightman. Vulnerability Inheritance Programmable Logic Controllers. *GreHack'13*, 2013.

[19] W. L. Luyben, B. D. Tyreus, and M. L. Luyben. *PlantwideProcess Control.* McGraw-Hill, 1998.

[20] M. Mahdian, R. P. McAfee, and D. Pennock. The secretary problem with a hazard rate condition. In *Internet and Network Economics*, pages 708–715. 2008.

[21] S. Mannan. *Lees' Loss Prevention in the Process Industries: Hazard Identification, Assessment and Control*, volume 1. Butterworth Heinemann, 2005.

[22] T. McAvoy and N. Ye. Base control for the Tennessee Eastman problem. *Computers & Chemical Engineering*, 18(5):383 – 413, 1994.

[23] T. McEvoy and S. Wolthusen. A Plant-Wide Industrial Process Control Security Problem. In *CIP V*, volume 367, pages 47–56. 2011.

[24] C. McIntyre. Using Smart Instrumentation. *Plant Engineering: online magazine*, 2011.

[25] S. McLaughlin, D. Pohly, P. McDaniel, and S. Zonouz. A Trusted Safety Verifier for Process Controller Code. In *NDSS 2014*, 2014.

[26] M. Naedele. An Access Control Protocol for Embedded Devices. In *IEEE International Conference on Industrial Informatics*, pages 565–569, 2006.

[27] N. L. Ricker. Tennessee Eastman Challenge Archive. `http://depts.washington.edu/control/LARRY/TE/download.html`. retrieved: May, 2013.

[28] N. L. Ricker and J. Lee. Nonlinear model predictive control of the Tennessee Eastman challenge process. *Comp. & Chem. Engineering*, 19(9):961 – 981, 1995.

[29] J. Rrushi. SCADA protocol vulnerabilities. In *Critical Infrastructure Protection*, volume 7130 of *LNCS*, pages 150–176. 2012.

[30] J. F. Smuts. *Process Control for Practitioners.* OptiControls Inc, 2011.

[31] U.S. Chemical Safety and Hazard Investigation Board. T2 Laboratories Inc. Reactive Chemical Explosion: Final Investigation Report. 2009.

[32] M.-K. Yoon and G. F. Ciocarlie. Communication Pattern Monitoring: Improving the Utility of Anomaly Detection for Industrial Control Systems. In *SENT 2014*, 2014.