

**Ahmad Reza Afkari Mehrabani**

PhD in Engineering Science

Independent Researcher and Inventor

Pasadena, California, USA

ORCID: 0000-0001-9817-5850

afkari2007@gmail.com

CYBER THREAT MODEL AND PROTECTION ARCHITECTURE FOR THE AUTOMATED CONTROL SYSTEM OF A WASTE-TO-ENERGY PLANT

Abstract. A control system for thermal waste conversion has to be protected even when the plant documentation cannot be fully disclosed. This paper builds a reproducible threat model for such a system and tests how three mechanisms affect control quality: message validation, a consistency check between two measurement channels, and local limits on setpoints. The proposed architecture separates remote access, an industrial demilitarized zone, the operator zone and the local control zone. An independent local measurement channel supports fallback control, and the safety instrumented system stays outside the authority of any cyber detector. Since no identified plant model is available, the quantitative part relies on a dimensionless first-order surrogate process. The main series has 720 runs: eight scenarios, three protection variants and 30 random seeds. With a constant bias injected into one channel, the mean absolute error fell from 0.13643 to 0.04140, and the share of samples outside the comparison band fell from 59.20% to zero. For a slow drift, the median delay to the first alarm was 31 model steps. Equal bias in both channels went undetected, and the combined variant then performed exactly like the baseline. A further 540 runs examined noise and threshold. At a fixed threshold of 0.04, raising the measurement noise from 0.005 to 0.02 increased the per-sample false alarm rate from zero to 15.42%. The rate depends on the ratio of threshold to noise and agrees with a simple Gaussian estimate. All figures describe the surrogate model only. They are not plant test results and do not certify functional safety. The practical outcome is a traceable link between threats, acceptance rules, fallback behavior and measurable control outcomes, plus a list of the data needed to move the procedure to a hardware test bench.

Keywords: cyber-physical system; industrial cybersecurity; waste-to-energy; threat model; measurement integrity; command validation; simulation.

INTRODUCTION

Problem statement. In an automated thermal process, a wrong measurement or an altered command acts on physical equipment. For a waste-to-energy plant the object of protection is therefore wider than access to the operator interface. It also covers the trustworthiness of temperature, pressure and flow signals and of actuator states. Blocking a suspicious connection has to be judged together with what happens to the loop when it loses feedback. NIST SP 800-82 Rev. 3 takes the same view of operational technology and weighs security against safety, reliability and timing [1].

The system considered here is the control layer of a gas production plant based on thermal conversion of waste. At the functional level it has four parts: material feed, a thermal reaction zone, gas treatment, and process control. Design details of the plant are subject to ongoing patent work and are not disclosed. The available design drawings also do not define transfer functions, the PLC cycle, sensor errors, communication protocols or time series. The plant dynamics cannot be identified from them. The research question is therefore narrower. What do joint checks of message origin, measurement consistency and command admissibility gain, and where do they fail, in an explicitly defined surrogate model?

Analysis of recent research and publications. IEC 62443-3-2 describes how to define the system under consideration, partition it into zones and conduits, and assess the related risks [2]. MITRE ATT&CK for ICS catalogues adversary behavior, including spoofed reporting messages that carry process data [3]. These sources give a structure for threats. They say nothing about control quality for a particular process. The OPC UA security model covers authentication, integrity and replay protection [4]. A valid cryptographic check, however, does not prove that the measured value is true when the source itself is compromised.



NISTIR 8219 applies behavioral anomaly detection in demonstration manufacturing environments [5]. Filonov et al. used time-series forecasting for early attack detection in the Tennessee Eastman process [6]. Hartung et al. compared modern unsupervised anomaly detection methods on data from the same process [7]. The ICS-Flow dataset combines network observations with process variables for testing detectors [8]. These works show different ways to build an experimental base. Their numbers cannot be carried over to a gasification plant without comparable data and a matching problem setup.

For local monitoring, the platform proposed by Grynkevych, Vasylenko and Rudin is relevant [9]. It combines Wazuh, event processing and on-premises analytics, and it deals with collecting and interpreting security events. In a process control loop, an event also has to be tied to the operating mode, and the effect of a response on the process has to be checked. Grynkevych et al. described firewall policy management as infrastructure-as-code, with a configuration repository, request validation and a staged rollout plan [10]. Both works are used here as architectural references for logging and for control of inter-zone policy. Their results do not replace a check of process dynamics.

One question remains open in the literature reviewed. How can threats be mapped onto the behavior of a local loop when no identified plant model exists? This paper offers a reproducible setup with three protection variants and a separate common-mode scenario. The comparison covers not only whether an alarm appears, but also which data the controller uses afterwards. This makes it possible to separate the effect of detection from the effect of fallback control.

The purpose of the article is to develop a cyber threat model and a protection architecture for the automated control system of a waste-to-energy plant, and to quantify a limited set of its mechanisms in a synthetic experiment. The objectives are: (1) to define assets and trust boundaries; (2) to formulate acceptance rules for data and commands; (3) to compare three control variants under eight scenarios; (4) to analyze sensitivity to measurement noise and detection threshold.

THEORETICAL FOUNDATIONS OF THE STUDY

System boundary and attacker capabilities. The functional boundary includes material feed, thermal conversion, gas treatment, local control and operator supervision. The list of measured quantities is a design proposal. The paper does not claim that specific sensors or remote access exist in the real plant. For digital signals, integrity, timeliness and availability are protected. Setpoints and configurations additionally need traceable changes and authorization checks.

Four attacker capabilities are considered: injection of messages without valid credentials; replay of previously captured messages; compromise of an authorized telemetry source; and misuse of legitimate access to setpoints. One scenario deliberately goes beyond the baseline trust boundary: coordinated tampering with both the primary and the local channel. In the main series the attacker cannot silently modify code in the trusted local controller. That possibility is treated as residual risk.

Table 1 links each asset to a threat and to the planned response. S1–S7 are scenario labels introduced in this paper, not MITRE technique IDs. The register is not a quantitative risk assessment. Attack frequencies and losses are unknown, so no probabilities or expected losses are assigned.

Table 1

Assets, scenarios and limits of the protective measures

Scenario	Asset and impact	Protection	Residual limitation
S1 Bias	Telemetry: constant offset	Comparison with the local channel	Requires a trusted second channel
S2 Drift	Telemetry: slow change	Discrepancy check	Changes below the threshold pass
S3 Replay	Freshness of data	Sequence number, session, message age	A compromised source can refresh metadata
S4 Command	Unauthorized actuator control	Authentication and operation permit	Does not stop every insider action
S5 Setpoint	Valid session, inadmissible setpoint	Local admissible region	Limits need process justification
S6 Dropout	No new telemetry	Timeout and local mode	Depends on the backup channel
S7 Common bias	Same error in both channels	Diversity, separate protection	Not detected by the studied scheme

Architecture and the functional safety boundary. The proposed architecture is shown in Fig. 1. An industrial demilitarized zone (DMZ) separates remote access from the operator zone, and the local loop does not depend on any cloud service. Every inter-zone flow is defined by its source, destination, permitted operation and direction. Engineering access is a separate role, and maintenance sessions are time-limited. The log records the subject, the old and new values, the reason and the result of the check. This is a design specification, not a description of a deployed network.

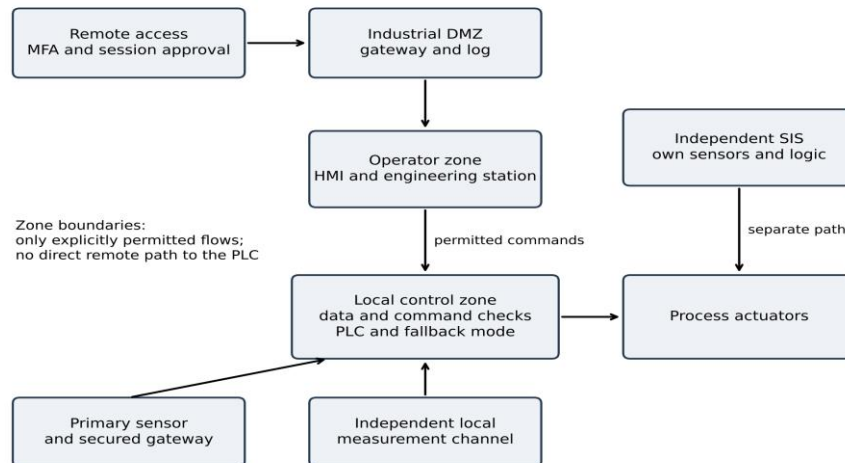


Fig. 1. Proposed trust zones and local control paths

Where devices support secure protocols, mutual authentication, message signing and certificate lifecycle management are advisable. OPC UA is one possible implementation, not a requirement [4]. For a legacy analog sensor, a secure gateway vouches for the origin of the digital message from the gateway. It says nothing about the physical correctness of the signal before it. Physical tampering calls for other measures: protected wiring, diagnostics and diverse measurements.

Signed updates, configuration version control and a tested recovery procedure complement channel protection. NIST SP 800-193 sets out principles for protecting firmware, detecting changes and recovering it [11]. How far they apply to a given PLC depends on the vendor. Network latency, cryptographic overhead, update verification and recovery time were not measured in this work.

The safety instrumented system (SIS) is shown as an independent loop. Its specification and lifecycle belong to functional safety [12]. A cyber detector output alone is not a reason for an emergency shutdown, because stopping feed, ventilation or cooling can have very different consequences. The safe state is defined by a process hazard analysis. The fallback mode in the model below is not an emergency control algorithm for a real reactor, and no SIL or SL levels are assigned.

Control of inter-zone policy changes. Segmentation keeps its value only if later changes are controlled. A firewall configured once does not prevent an overly broad rule from appearing during maintenance. For the proposed architecture, the permitted flows are kept as a versioned description, and preparing a change is separated from activating it. The approach follows policy-as-code practice [10] but adds coordination with the current process mode.

Each change request names the initiator, the source and destination zones, the required operation, the validity period and the reason for access. Before activation at least three conditions are checked. The rule must not open a direct path from the external network to the PLC. The service role must not gain the right to change local setpoint limits. The change must not interrupt critical measurements. The process owner approves the moment of activation. A syntax check is no substitute for this step, since a syntactically valid rule can still block a needed loop or open unnecessary access.

The procedure starts with a diff between the current and proposed configurations. The diff is tested on an isolated network segment, and an authorized person then approves activation. The version ID, checksum, check results and the last known good configuration are stored. Temporary access expires automatically, and an extension needs a new justification. If the link to the central repository is lost, the local system keeps the last approved policy. Loading an unverified configuration just to restore connectivity is not allowed.

Rollback also needs a process check. The previous network policy may no longer match the installed equipment or the active mode. A decision to roll back is therefore made after checking reachability of permitted



nodes and health of the control channels. This procedure is part of the design. Its duration, false rejection rate and effect on availability were not evaluated numerically.

Logging and justification of the protective response. Local event collection, as discussed in [9], gives a basis for incident investigation without sending logs to an external service. For the architecture studied here, a standard security record is extended with process context: plant mode, measurement ID, message number and age, reason for rejection, and the backup source actually used. For a command, the requested and accepted setpoints and the version of the rule applied are also stored. With such a record it is possible to reconstruct not only the alarm but also the decision that changed the control action.

The cause of an anomaly should not be inferred from sensor disagreement alone. The same pattern can come from tampering, a fault, a delay or bad calibration. In the proposed scheme the local rule only selects the data source for control. Diagnosing the cause is a separate task. The operator sees both channel values, the exceeded threshold and the message check status. Logins to the engineering station and configuration changes are correlated with the moment the discrepancy appeared. Such a match is a reason to investigate, not proof of an attack.

Logging must not delay the PLC cycle. Events are exported asynchronously through a bounded local buffer. A buffer overflow is logged as a loss of observability. It does not change setpoints and does not disable measurement checks. Synchronized timestamps support time correlation, and a monotonic counter in each source reveals gaps. If time synchronization can no longer be trusted, the exact order of events across nodes may stay uncertain.

The analytics component may group messages and prepare explanations for the operator. It has no right to write PLC outputs or SIS logic directly. If the analytics do not respond, local rules still run. This split lets monitoring grow without adding another software service to the critical control path. At the bench stage, loss of the log server, buffer overload and timestamp errors should be tested separately from measurement tampering scenarios.

METHODOLOGY OF THE STUDY

Acceptance rules for information flows. For a telemetry message m the check covers origin, membership in the active session, a monotonic sequence number q and message age. The general acceptance rule is given in (1). Here Auth is successful authentication, Session is a valid session, τ is the maximum message age, and q_{prev} is the number of the last accepted message. The metadata must be cryptographically bound to the payload. After a device restart a new authenticated session is required. Simply resetting the counter without it weakens protection.

$$V(m) = \text{Auth}(m) \wedge \text{Session}(m) \wedge (q > q_{\text{prev}}) \wedge (0 \leq t - t_m \leq \tau) \quad (1)$$

Cryptography is not simulated. Its outcome is a logical flag for the unauthorized command, and telemetry replays are represented by a repeated sequence number and timestamp. The experiment thus tests the consequences of the acceptance rules. It does not test the strength of the protocol, key storage or certificate handling. A real protocol would also need to handle packet reordering and the details of the transport session.

An additional predicate C compares the primary measurement y with the independent local measurement y_L . If the difference exceeds h , the primary value is rejected and the local value is used for control. This logic assumes that the local channel is the trusted one. Two sensors without such an assumption cannot tell which of them is wrong.

$$C(k) = |y(k) - y_L(k)| \leq h \quad (2)$$

A setpoint is accepted only inside the region R and only if the step from the previous accepted setpoint is limited. In a production system R depends on the mode, equipment state and process permits. The numerical experiment uses a fixed interval. If the rule is violated, the previous setpoint is kept. A user role check alone is not sufficient here.

$$A(r_k) = (r_k \in R) \wedge (|r_k - r_{\text{prev}}| \leq \delta) \quad (3)$$

Surrogate dynamics and compared variants. The process is a scalar first-order model (4). The variable x is a dimensionless state on a conventional scale. It is not a temperature, a gas fraction or an energy output, and it is not bounded to $[0, 1]$. The control action u is clipped to $[0, 1]$. A model step has no assigned duration in seconds. The coefficients were chosen to show the effect of information attacks and were not identified from the plant.

$$x(k+1) = 0.98x(k) + 0.02u(k) + d(k) + w(k) \quad (4)$$

$$u(k) = \text{clip}_{[0,1]} \{0.5 + 2[r(k) - y_{\text{used}}(k)]\} \quad (5)$$

The initial state is $x(0) = 0.5$ and the nominal setpoint is $r = 0.5$. The disturbance $d(k) = 0.002 \sin(k/80) + 0.003 \cdot I(300 \leq k < 650)$ is present in all scenarios, including the one without an attack, where I is the indicator function. It continues after the attack ends so that recovery under load can be observed. The process noise w is



normal with standard deviation 0.0005. The noises of the two measurements are independent, zero-mean and normal with $\sigma = 0.005$. Normal noise is unbounded, so the results do not prove invariance of a safe region.

Three variants are compared: B, N and P. Variant B uses whatever telemetry it receives. When telemetry is missing it holds the last value, it executes an injected command, and it does not limit the setpoint. Variant N rejects replays, stale data and commands that fail authentication. After three consecutive rejected measurements N applies $u = 0.5$ until acceptable data return; before that it uses the last accepted value. Variant P adds check (2), the limits $R = [0.45, 0.55]$ and $\delta = 0.02$, and switches to the local channel on the first rejected measurement. P has no fixed $u = 0.5$ during a dropout and keeps closed-loop local control instead. The main series uses $h = 0.04$ and a maximum message age $\tau = 2$ model steps.

N and P differ both in detection and in fallback policy. The gap between their results therefore cannot be credited to the threshold detector alone. The unauthorized command scenario shows the effect of message validation separately. The common-bias scenario tests how much the combined approach depends on its trust assumption.

Table 2 lists the scenarios of the main series. Each run has 900 steps, and the attack is active from step 300 to step 499 inclusive. At every step the generator draws the primary measurement noise, the local measurement noise and the process noise, in that order and regardless of the scenario. Seeds 0 to 29 are used with Python's `random.Random`. All variants therefore see identical noise realizations, which makes the comparison paired.

Table 2

Scenarios of the computational experiment

Code	Name in code	Intervention at steps 300–499
S0	normal	No attack; process disturbance kept
S1	bias	0.25 subtracted from the primary measurement; valid metadata
S2	drift	$\min[0.15; 0.001(k - 299)]$ subtracted; valid metadata
S3	replay	Value, number and time from step 299 repeated
S4	command	$u = 0.95$ injected; authentication flag false
S5	setpoint	Authorized setpoint changed to $r = 0.85$
S6	dropout	No primary telemetry
S7	common_bias	0.25 subtracted from both measurements

Metrics and sensitivity analysis. Control quality is measured on the true model state, which the detector cannot see, over a fixed window $W = \{300, \dots, 649\}$. The window covers the attack and 150 steps after it. The metrics are the mean absolute error MAE and the share Q of steps outside the band $|x - 0.5| \leq 0.1$. The band is a comparison criterion for this study, not a process alarm limit.

$$\text{MAE} = \frac{1}{350} \sum_{k \in W} |x(k) - 0.5| \quad (6)$$

$$Q = \frac{100}{350} \sum_{k \in W} \mathbb{I}(|x(k) - 0.5| > 0.1) \quad (7)$$

The delay D is the step of the first alarm inside the attack interval minus 300. $D = 0$ means a response in the same discrete step, not zero hardware latency. If there is no alarm, the delay is undefined. The false alarm rate FPR is the number of alarmed steps divided by the number of steps in fully nominal runs (S0). It is a per-sample rate, not an incident rate.

The main series contains $8 \times 3 \times 30 = 720$ runs. A separate series of $3 \times 3 \times 2 \times 30 = 540$ runs of variant P covers $\sigma \in \{0.005; 0.01; 0.02\}$, $h \in \{0.02; 0.04; 0.08\}$ and the normal and drift scenarios. Thresholds were fixed before these runs, and no parameters were fitted on test data. Means and sample standard deviations describe variation over 30 noise realizations. They do not describe uncertainty in the properties of an industrial plant.

RESULTS OF THE STUDY

Comparison of protection mechanisms. Table 3 gives the results of the main series: mean MAE for all three variants, mean Q for B and P, and the median delay for P. “n/a” means no detection or a metric that does not apply. P detected scenarios S1–S6 in all 30 runs and S7 in none. The 30/30 count reflects these specific attacks. It is not an estimate of detection probability for an arbitrary attack.



Table 3

Metrics over 30 realizations of each scenario

Code	MAE B	MAE N	MAE P	Q B, %	Q P, %	D P, steps
S0	0.04139	0.04139	0.04139	0.00	0.00	n/a
S1	0.13643	0.13643	0.04140	59.20	0.00	0
S2	0.07724	0.07724	0.04286	21.79	0.00	31
S3	0.07237	0.06015	0.04140	10.71	0.00	0
S4	0.28029	0.04139	0.04139	67.12	0.00	0
S5	0.17346	0.17346	0.04139	61.61	0.00	0
S6	0.07237	0.06015	0.04140	10.71	0.00	0
S7	0.13643	0.13643	0.13643	59.20	59.20	n/a

Without an attack all variants give the same MAE of 0.04139. The nonzero error comes from the disturbance and from the lack of an integral term in the controller. For S1, B and N have a mean MAE of 0.13643 (SD 0.00037), and P has 0.04140 (SD 0.00037). The relative reduction is about 69.66%. It reflects the availability of an intact local channel in this model. It does not show a similar drop in the risk of a real accident.

Bias S1 passes the checks of N because the source keeps valid metadata. The unauthorized command S4, by contrast, is already blocked by N, and MAE falls from 0.28029 to 0.04139. For S5 authentication is not enough: only P rejects the setpoint through the local admissibility rule. Command protection therefore has to check both who issues the action and what the action is.

For S3 and S6 variant B behaves the same way. Replaying the last value and holding it give the same feedback. Under N, MAE is 0.06015 and mean Q is 0.32%. Under P the values are 0.04140 and 0%. The advantage of P here comes from local feedback, not from better replay detection, since N also raises an alarm at the first step. The fixed fallback action of N cannot compensate for a changing external load.

Under drift S2 the median delay of P is 31 steps. Part of the altered measurements is accepted before the threshold is crossed. After the crossing, noise can push the difference back below the threshold, because the scheme has no hysteresis or alarm latching. The first alarm therefore does not mean that the whole interval is flagged. Mean MAE is 0.04286 for P and 0.07724 for B and N.

Fig. 2 shows trajectories for seed 0. On the left, P holds control quality while one channel is biased. On the right, all three variants follow the same path: common bias S7 keeps the two channels consistent with each other. The B and N curves overlap, and on the right the P curve overlaps them too. The gray area marks the attack, and the horizontal line at 0.6 is the upper edge of the comparison band.

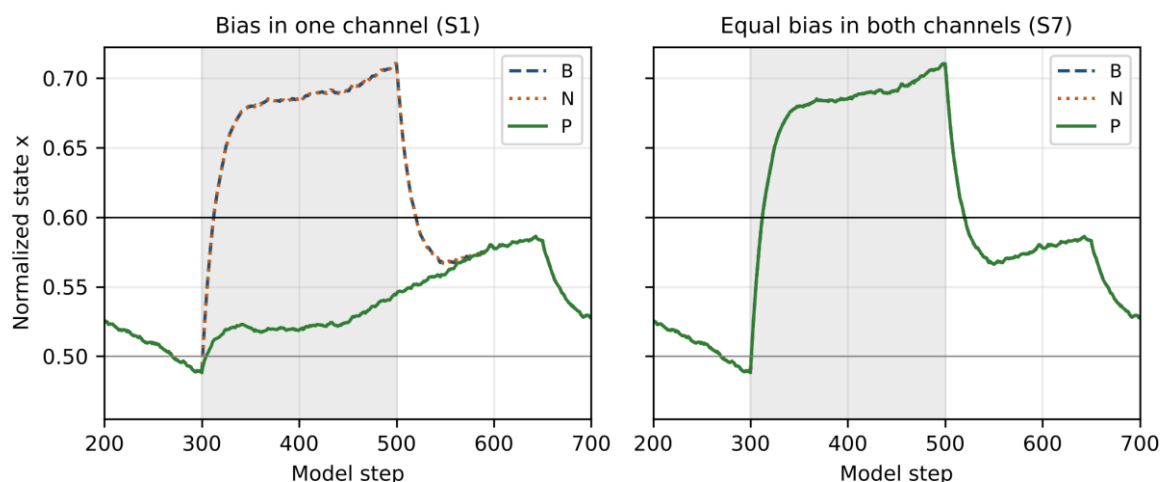


Fig. 2. Model trajectories under single-channel and common bias

For S7 the mean MAE of P stays at 0.13643, and Q is 59.20%. This negative result matters. Two channels compromised in the same way do not provide independent evidence. Protection can be strengthened with different measurement principles, separate power supplies and administration, protection of the primary transducer, and independent physical limits. Each of these needs its own evaluation. Adding a third sensor does not by itself remove a common cause of failure.



Sensitivity to noise and threshold. Table 4 compares the false alarm rate on 27,000 nominal samples for each parameter pair with the median first-alarm delay over 30 drift runs. The dynamic model and the attack amplitude stay the same in all cases. The last column gives the theoretical per-sample rate discussed below.

Table 4

Trade-off between false alarms and delay in variant P

σ	h	FPR S0, %	D S2, steps	MAE S2	FPR theory, %
0.005	0.02	0.44	12.0	0.04175	0.47
0.005	0.04	0.00	31.0	0.04286	< 0.01
0.005	0.08	0.00	73.0	0.04740	< 0.01
0.010	0.02	15.42	3.0	0.04167	15.73
0.010	0.04	0.44	23.5	0.04284	0.47
0.010	0.08	0.00	60.0	0.04724	< 0.01
0.020	0.02	47.44	1.0	0.04146	47.95
0.020	0.04	15.42	3.0	0.04246	15.73
0.020	0.08	0.44	34.0	0.04688	0.47

At $\sigma = 0.005$, lowering h from 0.04 to 0.02 cuts the median drift delay from 31 to 12 steps but raises FPR from 0% to 0.44%. At $h = 0.04$, raising σ to 0.02 pushes FPR to 15.42%. The short delay under high noise is partly a product of frequent background alarms, not of better attack recognition. Comparing first-alarm times alone would give a false impression of improvement.

For independent, identically distributed Gaussian errors, the difference of two nominal measurements has standard deviation $\sqrt{2}\sigma$. The probability of exceeding the threshold in one sample is then $2[1 - \Phi(h/(\sqrt{2}\sigma))]$, where Φ is the standard normal distribution function. The rate depends only on the ratio h/σ . This explains why the pairs with equal ratio in Table 4 give identical observed FPR: the seeds are the same and the noise is simply scaled. For $h/\sigma = 2$ the estimate is 15.73%, against 15.42% observed. For $h = 0.04$ and $\sigma = 0.005$ the estimate is very small but not zero. The absence of false alarms in a finite sample does not prove that they cannot occur.

Analytical explanation of the bias error. The effect of bias can be traced without further simulation. Assume no noise, a constant disturbance d , and a primary channel that reports $y = x - b$, where $b > 0$ is an underestimate. With setpoint 0.5 and an unsaturated controller, substituting (5) into (4) gives $x(k+1) = 0.94x(k) + 0.03 + 0.04b + d$. The coefficient 0.94 ensures convergence. The equilibrium, however, moves to $x^* = 0.5 + (2/3)b + d/0.06$. A stable loop can thus hold an unwanted state when it is fed false data.

For $b = 0.25$ and $d = 0$ the equilibrium shift is 0.1667. This explains the direction of the S1 and S7 trajectories but does not predict their peaks exactly, because in the simulation d varies over time and noise is present. When control switches to the intact local channel, the term with b disappears. If both channels carry the same bias, their difference does not contain b , so the consistency check cannot see it. This follows from the structure of the measurements, not from a poor choice of threshold.

If only the primary channel is biased, without noise and with $|b| \leq h$, the discrepancy never exceeds the threshold. In the same unsaturated model the additional steady-state error is then at most $2h/3$, about 0.0267 normalized units for $h = 0.04$. This bound applies only to undetected constant bias under the stated assumptions. A real process with several states, delays, saturation or correlated errors would need a different derivation.

Limits of applicability and further validation. The results come from a low-dimensional example, not from a thermochemical gasification model. The example has no temperature-pressure coupling, variable feedstock, gas analysis, transport delays, actuator rate limits, start-up and shutdown modes, or faults in auxiliary equipment. The attack parameters were chosen to test specific properties and do not represent the real frequency or strength of cyber incidents.

The architecture is broader than the numerical test. Segmentation, multi-factor authentication, log protection, updates and the SIS are not implemented in the model. No network attacks on hardware, cryptographic measurements or comparison with a trained machine learning detector were carried out. The work therefore does not show superiority over the methods in [6]–[8] and does not demonstrate compliance with any standard. Its conclusions are limited to the effect of the chosen rules on control in the given scenarios.

The next stage should start with an agreed list of signals, their accuracy and sampling rates, the actual communication channels and the permitted operating modes. On a bench with a hardware controller, detection, fallback switching and process impact should be tested separately, with end-to-end latency measured from signal arrival to output change. Thresholds and time windows should be set on nominal data and evaluated on held-out data. That data should include process disturbances without attacks, failure of the local channel and common-cause failures.



Engineering deployment also requires a check of legitimate setpoint changes. The fixed interval in this model is deliberately narrow and could reject useful operator actions in another mode. The policy for leaving fallback mode should require confirmation of data quality, a discrepancy check and operator approval. Tests on an operating plant are acceptable only under an approved program that excludes hazardous interference with the running process.

CONCLUSIONS AND FUTURE RESEARCH

A threat model was built for a generalized control system of a waste-to-energy plant, together with an architecture that links network trust boundaries to local checks of data and commands. The procedure keeps three things apart: message authenticity, physical consistency of measurements, and process admissibility of an action. Its contribution is a verifiable mapping between these mechanisms, not a new cryptographic protocol.

In 720 synthetic runs, message validation stopped the unauthorized command but did not stop bias from an authorized source. With a trusted local measurement, the combined variant reduced MAE under constant bias from 0.13643 to 0.04140. Common bias in both channels removed this advantage completely. A further 540 runs showed how false alarms depend on noise and threshold, in line with the Gaussian estimate. All results were reproduced by an independent re-implementation of the model from the description in the Methodology section.

The scenarios and code can serve as a starting point for a bench test program. Further work should cover multivariable dynamics, mode-dependent thresholds, common-cause failures of measurement channels and hardware latency measurements. Deployment will need the parameters of the specific plant, verified independence of measurements and agreed process limits.

Declaration on the use of generative AI. Generative AI tools (ChatGPT, OpenAI; Claude, Anthropic) were used for language editing, drafting of the simulation code and preparation of figures. The author reviewed all text, code and results and takes full responsibility for the content of the article.

Data availability. All numerical data are synthetic. The simulation code and individual run results are available from the author on request. Design drawings and other confidential materials related to the plant are not included.

REFERENCES

1. Stouffer, K., Pease, M., Tang, C., Zimmerman, T., Pillitteri, V., Lightman, S., Hahn, A., Saravia, S., Sherule, A., & Thompson, M. (2023). *Guide to operational technology (OT) security* (NIST SP 800-82 Rev. 3). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-82r3>
2. International Electrotechnical Commission. (2020). *Security for industrial automation and control systems. Part 3-2: Security risk assessment for system design* (IEC 62443-3-2:2020). <https://webstore.iec.ch/en/publication/30727>
3. MITRE. (n.d.). *Reporting message (T1692.002)*. MITRE ATT&CK for ICS. Retrieved October 15, 2026, from <https://attack.mitre.org/techniques/T1692/002/>
4. OPC Foundation. (n.d.). *OPC unified architecture. Part 2: Security model*. Retrieved October 15, 2026, from <https://reference.opcfoundation.org/specs/OPC-10000-2/full>
5. McCarthy, J., Powell, M., Stouffer, K., Tang, C., Zimmerman, T., Barker, W., Ogunyale, T., Wynne, D., & Wiltberger, J. (2020). *Securing manufacturing industrial control systems: Behavioral anomaly detection* (NISTIR 8219). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.IR.8219>
6. Filonov, P., Kitashov, F., & Lavrentyev, A. (2017). *RNN-based early cyber-attack detection for the Tennessee Eastman process* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.1709.02232>
7. Hartung, F., Franks, B. J., Michels, T., Wagner, D., Liznerski, P., Reithermann, S., Fellenz, S., Jirasek, F., Rudolph, M., Neider, D., Leitte, H., Song, C., Kloepper, B., Mandt, S., Bortz, M., Burger, J., Hasse, H., & Kloft, M. (2023). *Deep anomaly detection on Tennessee Eastman process data* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2303.05904>
8. Dehlaghi-Ghadim, A., Helali Moghadam, M., Balador, A., & Hansson, H. (2023). *Anomaly detection dataset for industrial control systems* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2305.09678>
9. Grynkevych, G., Vasylenko, V., & Rudin, D. (2026). Local open-source platform for cyber incident monitoring and response with AI support as an alternative to EDR/XDR solutions. *Cybersecurity: Education, Science, Technique*, 1(33), 234–241. <https://doi.org/10.28925/2663-4023.2026.33.1211>
10. Grynkevych, G., Vasylenko, V., Rudin, D., & Pliukhov, O. (2026). Design and implementation of automated firewall policy management in enterprise telecommunication systems using infrastructure-as-code. *Zviazok*, (4), 3–12. <https://doi.org/10.31673/2412-9070.2026.049412>



11. Regenscheid, A. (2018). *Platform firmware resiliency guidelines* (NIST SP 800-193). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-193>
12. International Electrotechnical Commission. (2017). *Functional safety. Safety instrumented systems for the process industry sector. Part 1: Framework, definitions, system, hardware and application programming requirements* (IEC 61511-1:2016+AMD1:2017 CSV). <https://webstore.iec.ch/en/publication/61289>

**Ахмад Реза Афкари Мехрабани**

Доктор філософії (PhD) з інженерних наук

Незалежний дослідник і винахідник

Пасадена, Каліфорнія, США

ORCID: 0000-0001-9817-5850

afkari2007@gmail.com

**МОДЕЛЬ КІБЕРЗАГРОЗ ТА АРХІТЕКТУРА ЗАХИСТУ АВТОМАТИЗОВАНОЇ СИСТЕМИ
КЕРУВАННЯ УСТАНОВКОЮ ПЕРЕТВОРЕННЯ ВІДХОДІВ НА ЕНЕРГІЮ**

Анотація. Систему керування установкою термічного перетворення відходів потрібно захищати навіть тоді, коли технічну документацію неможливо розкрити повністю. У статті побудовано відтворювану модель загроз для такої системи і перевірено, як три механізми впливають на якість керування: перевірка повідомлень, контроль узгодженості двох вимірювальних каналів і локальне обмеження уставок. Запропонована архітектура розділяє зовнішній доступ, промислову демілітаризовану зону, зону оператора та локальну зону керування. Незалежний локальний вимірювальний канал забезпечує резервне керування, а система протиаварійного захисту залишається поза повноваженнями кібердетектора. Оскільки ідентифікованої моделі установки немає, кількісну частину виконано на безрозмірній сурогатній моделі першого порядку. Основна серія містить 720 запусків: вісім сценаріїв, три варіанти захисту і 30 початкових значень генератора випадкових чисел. За сталого зміщення одного каналу середня абсолютна похибка зменшилася з 0,13643 до 0,04140, а частка відліків поза смугою порівняння зменшилася з 59,20% до нуля. За повільного дрейфу медіана затримки першого сигналу становила 31 крок моделі. Однакове зміщення обох каналів не було виявлено, і комбінований варіант у цьому випадку не відрізнявся від базового. Додаткові 540 запусків стосувалися шуму та порога. За фіксованого порога 0,04 збільшення шуму вимірювань з 0,005 до 0,02 підвищило частку хибних сигналів з нуля до 15,42%. Ця частка залежить від відношення порога до шуму і узгоджується з простою гауссівською оцінкою. Усі числа характеризують лише сурогатну модель. Вони не є результатами випробувань установки і не підтверджують функціональну безпеку. Практичним результатом є простежуваний зв'язок між загрозами, правилами прийняття даних, резервною поведінкою та вимірюваними показниками керування, а також перелік даних, потрібних для перенесення процедури на апаратний стенд.

Ключові слова: кіберфізична система; промислова кібербезпека; перетворення відходів на енергію; модель загроз; цілісність вимірювань; перевірка команд; імітаційне моделювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Stouffer, K., Pease, M., Tang, C., Zimmerman, T., Pillitteri, V., Lightman, S., Hahn, A., Saravia, S., Sherule, A., & Thompson, M. (2023). *Guide to operational technology (OT) security* (NIST SP 800-82 Rev. 3). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-82r3>
2. International Electrotechnical Commission. (2020). *Security for industrial automation and control systems. Part 3-2: Security risk assessment for system design* (IEC 62443-3-2:2020). <https://webstore.iec.ch/en/publication/30727>
3. MITRE. (n.d.). *Reporting message (T1692.002)*. MITRE ATT&CK for ICS. Retrieved October 15, 2026, from <https://attack.mitre.org/techniques/T1692/002/>
4. OPC Foundation. (n.d.). *OPC unified architecture. Part 2: Security model*. Retrieved October 15, 2026, from <https://reference.opcfoundation.org/specs/OPC-10000-2/full>
5. McCarthy, J., Powell, M., Stouffer, K., Tang, C., Zimmerman, T., Barker, W., Ogunyale, T., Wynne, D., & Wiltberger, J. (2020). *Securing manufacturing industrial control systems: Behavioral anomaly detection* (NISTIR 8219). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.IR.8219>
6. Filonov, P., Kitashov, F., & Lavrentyev, A. (2017). *RNN-based early cyber-attack detection for the Tennessee Eastman process* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.1709.02232>
7. Hartung, F., Franks, B. J., Michels, T., Wagner, D., Liznerski, P., Reithermann, S., Fellenz, S., Jirasek, F., Rudolph, M., Neider, D., Leitte, H., Song, C., Kloepper, B., Mandt, S., Bortz, M., Burger, J., Hasse, H., & Kloft, M. (2023). *Deep anomaly detection on Tennessee Eastman process data* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2303.05904>



8. Dehlaghi-Ghadim, A., Helali Moghadam, M., Balador, A., & Hansson, H. (2023). *Anomaly detection dataset for industrial control systems* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2305.09678>
9. Grynkevych, G., Vasylenko, V., & Rudin, D. (2026). Local open-source platform for cyber incident monitoring and response with AI support as an alternative to EDR/XDR solutions. *Cybersecurity: Education, Science, Technique*, 1(33), 234–241. <https://doi.org/10.28925/2663-4023.2026.33.1211>
10. Grynkevych, G., Vasylenko, V., Rudin, D., & Pliekhov, O. (2026). Design and implementation of automated firewall policy management in enterprise telecommunication systems using infrastructure-as-code. *Zviazok*, (4), 3–12. <https://doi.org/10.31673/2412-9070.2026.049412>
11. Regenscheid, A. (2018). *Platform firmware resiliency guidelines* (NIST SP 800-193). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-193>
12. International Electrotechnical Commission. (2017). *Functional safety. Safety instrumented systems for the process industry sector. Part 1: Framework, definitions, system, hardware and application programming requirements* (IEC 61511-1:2016+AMD1:2017 CSV). <https://webstore.iec.ch/en/publication/61289>

Отримано редакцією журналу / Received: 22.08.26

Прорецензовано / Revised: 31.08.26

Схвалено до друку / Accepted: 26.10.26



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.