Redundancy Link Security Analysis: An Automation Industry Perspective

Björn Leander*, Bjarne Johansson*†, Saad Mubeen†, Mohammad Ashjaei†, Tomas Lindström*
* ABB Process Automation, Process Control Platform, Västerås, Sweden,
{bjorn.leander, bjarne.johansson, tomas.lindstrom}@se.abb.com
† Mälardalen University, Västerås, Sweden, {bjarne.johansson, saad.mubeen, mohammad.ashjaei}@mdu.se

***Abstract*—Industrial automation and control systems are responsible for running our most important infrastructures, providing electricity and clean water, producing medicine and food, along with many other services and products we take for granted. The safe and secure operation of these systems is therefore of great importance. Reliability is a fundamental requirement, with spatial controller redundancy being one important fault-tolerance mechanism. In current control system solutions, controller redundancy is implemented using state and heartbeat transfer communicated over dedicated physical links, based on an assumption of implicit trust. However, with the emerging network-centric control strategies there is a desire to transition into dynamic redundancy scenarios, where such dedicated links are unfeasible. In this paper, an approach for a threat-model based security analysis is presented from the perspective of the automation industry. The approach is applied to the communication links used for supporting control system redundancy within a network-centric architecture.**

I. Introduction

Industrial Automation and Control Systems (IACS) control and supervise a wide range of industrial systems, including power generation, mining, process industries, etc. Reliability is one of the main objectives for IACS, to allow disturbance free, safe and secure operations of controlled industrial processes. In a control system, a common mechanism for increasing reliability of essential function is to use *redundancy*, i.e., to duplicate and distribute the essential function over several components. This can be done with software as well as hardware components, and includes, e.g., controller redundancy schemes [1] and network redundancy schemes such as PRP [2] and MRP [3].

Many redundancy schemes are based on a primary service performing the actual work, with one or more backup services ready to take over in case of primary service failures. This setup is commonly referred to as warm standby redundancy [4]. To allow the transition from backup to primary, a failure detection mechanism is required, often implemented using a *heartbeat* or health-information link. For a minimum switchover time the backup must be in warm or hot standby, which requires knowledge of the current state of the primary, i.e., current output and input signals and other application program states, which the current control applications are, what part of the application is executing, etc. In practice, the primary checkpoints the internal states needed for the backup to assume the primary role. The checkpointing is implemented by the primary sending *state data* to the designated backup units, which allows them to keep track of the process state, enabling switchover with very short delay.

In current IACS, the state transfer and heartbeat has been exchanged over dedicated physical links and using proprietary protocols such as RMX by Emerson [5]. However, as a consequence of emerging design strategies such as Network-Centric Control (NCC) [6], the flexible redundancy schemes of the future cannot rely on dedicated links.

With the growing complexity and amount of interconnections within Operation Technology (OT) networks, along with an increasingly hostile cyber environment in general [7], [8], a shift of security model is required, from implicit trust towards zero-trust [9]. The Zero-trust model assumes any interaction could be malicious, which implies a check-all, trust no-one approach.

Problem statement: State and heartbeat data are essential aspects of many controller redundancy schemes. The currently used solutions are based on dedicated physical links and implicit trust between components. Emerging control system architectures require more flexible redundancy schemes, and operate in networks where a zero-trust security model is applicable, rendering current solutions potentially vulnerable.

Research objective and contributions: Given the motivation and problem descriptions above, it is essential to study and investigate the potential cybersecurity issues in such an architecture. Therefore, in this study we aim to describe and understand the potential cybersecurity threats particularly to heartbeat and state data links. Moreover, we present suggestions on mechanisms for countering the identified threats. The main contributions of this paper are:

- comparison between controller redundancy for traditional- and network-centric architectures;
- a methodology for threat-model based security analysis; and
- a detailed security analysis of the redundancy data links.

The paper outline is as follows. Related work is discussed in section II. Design of redundancy links, along with high-level requirements and an architecture comparison is provided in section III. The security analysis methodology is described in section IV, followed by the security analysis of the redundancy links in section V. Implications of the results are discussed in section VI and in section VII results are summarized and possible future directions of the work are outlined.

II. Background and Related Work

Standby redundancy, i.e., hardware duplication with one primary and one backup controller, represents the prevalent IACS redundancy deployment today [4], [10], [11].

With the advent of network-centric controllers, the potential for more flexible redundancy alternatives arises [6]. Hegazy et al. propose a potential cost reduction by employing cloud-based backups to reduce the hardware footprint and thereby reduce costs [12]. Another utilization of the network-centric controller is to combine controller redundancy with orchestrator and containerized controllers to circumvent redundancy deterioration [13]. While these works highlight the advantages of utilizing network-centric controllers, they do not specifically tackle the security challenges associated with these controllers. This is the main focus of this paper.

The network's reliability is crucial for the functionality of a distributed system, such as a redundant controller. Previous research has investigated fault-tolerance techniques in industrial Ethernet-based systems, as well as the real-time and reliability properties of industrial Ethernet protocols [14], [15]. Within this context, these studies concentrate on reliability rather than security. Our work focuses on the security of the link carrying specific types of information. Specifically, we focus on the information necessary to uphold the operational standby redundancy.

Using threat modeling as security analysis tool is of growing interest also in the industrial systems domain. It is stated in the IEC 62443 standard, part 4-1 [16], [17], as a Secure Development Life Cycle process requirement (SR-2). The analysis provided in our paper is influenced by the work of Khan *et al.* [18], which contains a threat model from the micorgrid domain. There are several other works that utilize the threat modeling in similar areas, e.g., on pump control [19] and on the OPC UA protocol [20].

Furthermore, there exist a few works that cover security perspectives of controller redundancy. Ma *et al.* [21] describe how an attacker can make use of weaknesses in redundancy architectures to perform hard to discover attacks. Preschern *et al.* [22] discuss the need for security enhancement of redundancy architectures. Sepehrzadeh et al. [23] on the other hand discuss how redundant components can enhance the security of a Cyber Physical System (CPS).

The existing works are related to threat modeling of industrial systems or consider some security perspectives of redundancy. To the best of our knowledge, none of the previous works describe and analyze the security of controller redundancy communication links, as is done in this paper.

III. Redundancy link design

As a basis for the security analysis, this section describes the design of the redundancy links as needed for warm controller standby. A comparison between traditional and a NCC-based architectures for redundancy is provided, followed by a discussion on requirements and a detailed description of the heartbeat and state transfer links respectively.

A. Architecture comparison

The current redundancy schemes are typically implemented with state and heartbeat being transferred over a dedicated link, as illustrated in Fig. 1a. This provides very good isolation properties, as the link is dedicated to redundancy, and no intermediate nodes or switches are needed. However, the architecture lacks flexibility, and the controller hardware must be compatible to allow the redundancy link's physical connection.

Using the emerging network-centric design strategy, as illustrated in Fig. 1b, there is no dedicated link for the redundancy data exchange, instead a common logical network[1] is used for all interactions between control services in the redundancy set, and the Field Communication Interfaces (FCI). This allows for more flexible deployment scenarios, e.g., with several backup nodes in a redundancy set (1ooN-redundancy), one hardware node potentially serving as backup for several nodes, etc. This is possible because the state is transferred using standardized network technologies, e.g., switched Ethernet.

B. Functional requirements

With the design change from static redundancy using dedicated link to flexible redundancy over shared link, some fundamental characteristics are changed. The basic functional requirements on redundancy characteristcs however stays the same. Deterministic, hard real-time failure detection and role selection are essential. The backup(s) must detect a primary's failure within a bounded and guaranteed timeframe. The failure detection and role selection mechanisms are crucial to enable a backup to discern when to resume the primary's operations. In configurations with one primary and one backup, the role selection is inherently linked to failure detection: a failed primary implies that the backup should adopt the primary role.

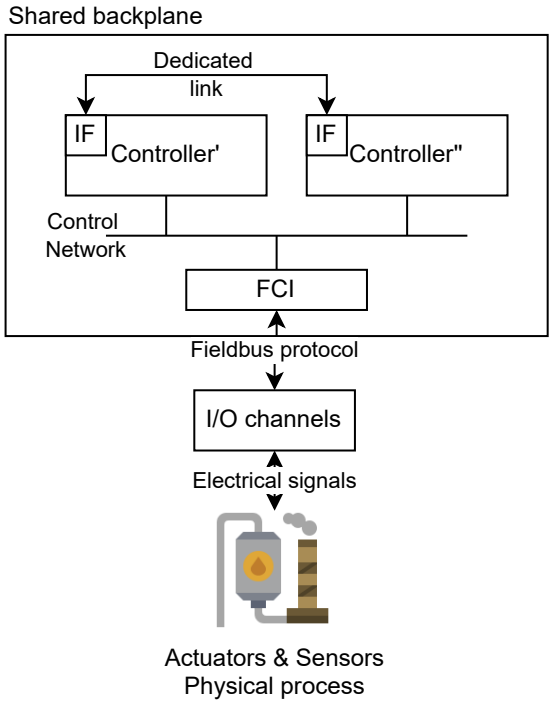
Deterministic, hard real-time, state transfer is vital for the backup to resume operations with the most recently externally exposed state. By doing so, the backup, transitioning to a primary, avoids communicating historical or outdated signal values to the controlled process.

The essence of this system lies in the failure detection and state transfer processes, which necessitate, as noted, hard real-time communication capabilities. From the receiver's perspective, the absence or delay of messages might be indistinguishable from a failure of the sending primary [24].

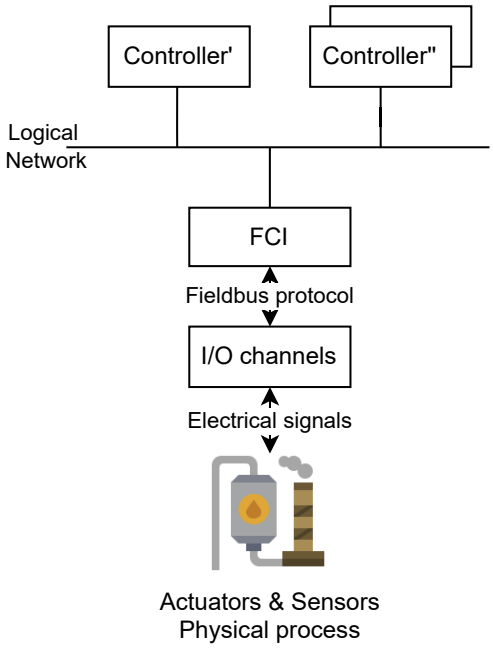
C. Failure detection and role selection

There are several algorithms for role selection, i.e., the mechanism for deciding on which of the instances are primary and which acts as backup. The used implementation follows the heartbeat bully algorithm, as presented in [25]. In this work however, the choice of algorithm for role selection is not very important, focus is on the reliability and availability features of the link used for communicating the role selection data.

[1]Common logical network: from the network users perspective it looks like one network, but the physical infrastructure may be more complex, e.g., defined using Software Defined Network (SDN).



(a) Architecture using a dedicated redundancy link for state and heartbeat transfer.



(b) Architecture for flexible redundancy in a network-centric design.

Fig. 1: Example architectures based on controller- and network-centric design respectively.

D. State transfer

Several strategies exist to allow a swift state take over, which all have different properties and requirements on the information needed by the backup to take over.

The choice of strategy used in this architecture is warm standby, which means that the backup is effectively executing the exact same logic as the primary, with the only exception that it does not set output-values. To be able to do this, the backup needs to know the signal values and system state as perceived by the primary service - making the state transfer a critical additional link of the redundancy protocol.

IV. A Threat Modeling based Security Analysis

The methodology used for the security analysis is a developed variant of threat modeling. Threat modeling is performed to identify and evaluate security threats to a system. The approach that is used in this work is inspired by the work by Khan *et al.* [18] with a few adaptations, and is outlined below:

1) *Enumerating threat consequences*: The first step of the analysis is to enumerate the consequences that different threats can potentially have, i.e., what are the things an attacker may want to achieve when targeting this specific system. The consequences are evaluated for criticality, to help prioritizing. There may be a need to revisit this step during the consequent ones to complement unforeseen consequences.
2) *Enumerating threat actors:* Looking at the security context of the system, different categories of threat actors or potential attackers can be identified. This is useful to consider, as different categories of attackers may be countered using different compensating measures, e.g., physical access to equipment can be mitigated by locked cabinets and other types of physical security mechanisms. Some attacker categories may be decided not to handle, due to the decided security ambition of the system and the level of sophistication of the attacker.
3) *Constructing a Data-Flow diagram (DFD):* A Data-Flow Diagram (DFD) is constructed for the system, including processes (P), data-flows (DF), data stores (DS) external actors (E) and security zones. The entities are all given unique identities.
4) *Define interactions to analyze:* Each of the data flows crossing a trust boundary are analyzed individually. Notation used for one boundary crossing is sender:data-flow:receiver, e.g., (P1:DF2:E5) for the interaction from process P1 to external entity E5 over data-flow DF2.
5) *Analyze and categorize threats:* For each zone-crossing, threats are enumerated and categorized according to the STRIDE model [26], which is an mnemonic for: **S**poofing, **T**ampering, (non-)**R**epudiation, **I**nformation disclosure, **D**enial of Service and **E**levation of privilege. Each threat is given an identity, and the threat actors that could possibly effectuate the threat are listed.
6) *Enumerate mitigations:* At this stage, mechanisms for threat mitigations are evaluated against the threats. A set of mechanisms is selected, based on the security ambition

of the system, and associated with respective threat ID.

7) *Evaluate residual risks:* Typically, a mitigation cannot entirely remove the threat, only limit the likelihood of its materialization, unless the mitigation is to entirely remove the component or interaction from which the threat arises. The risk assessment [27] is business-specific, but is usually calculated using a combination of the consequence of the threat and the likelihood of its materialization. The likelihood metric is always a difficult number to establish, and is sometimes left out of the analysis entirely.

The difference in this approach as compared to, e.g., the one of Khan [18], is the addition of threat actors and mitigations as part of the analysis. Putting threats, consequences, threat actors, and mitigations in the same artifact allows for a more powerful and transparent analysis. For instance, if deciding a specific threat actor should not be considered, that will automatically mean any enumerated threats or mitigations that are only associated with that actor could be discarded.

V. Security Analysis of Redundancy Links

In this section, a security analysis of the redundancy links is performed by utilizing the method outlined in the previous section, with exception of the residual risk analysis, which is out of scope for this paper.

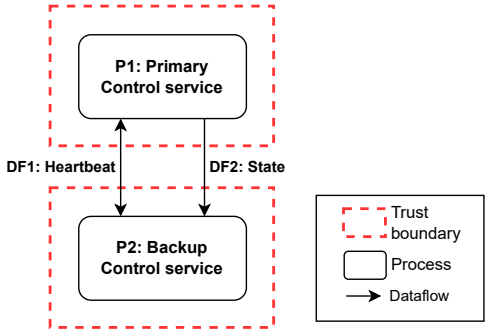


Fig. 2: Redundancy link data flow diagram.

A. Threat Consequences

The analysis provided in this paper is application agnostic, therefore only the consequences related to the redundancy function in itself can be described. The severity of the consequences are rated *Low*, *Medium*, *High*, and *Catastrophic* on a progressive scale. The following potential threat consequences are anticipated:

TC1 *Dual Primary:* Two controllers acting as primaries with both having access to the I/O, will lead to a potentially nondeterministic system behavior [24], and is a threat to both integrity and availability. ⇒ *Catastrophic*

TC2 *No Primary:* No controller is driving the process, also known as *loss of control*. The I/O values will behave in some predefined way: freeze, return to zero, or, if safety-instructed functionality (SIF), return to a predefined safe state. Threat to availability. ⇒ *High*

TC3 *Loss of redundancy:* The primary will keep running, but in the case of a failure or decreased health, no switchover will occur. ⇒ *High*

TC4 *Disclosure of state-related secrets:* The state data being transferred to the backup unit(s) in the redundancy set is application specific, and may contain sensitive information. ⇒ *Low-High*

TC5 *Incorrect state data:* The backup unit is (unknowingly) using incorrect checkpoint data. ⇒ *Medium*

TC6 *Disclosure of network-related information:* The redundancy settings and device health used in the system, i.e., which nodes are running as primaries, which are running as backup, is the information that could be useful in reconnaissance information for further attacks. ⇒ *Low*

TC7 *Changed output:* An output signal being set according to the attackers choice is a potential loss of safety. ⇒ *Catastrophic*

TC8 *Switchover:* A change of primary is unnecessarily triggered, i.e., the attacker can choose which control service is the primary. ⇒ *Medium*

B. Attacker classification

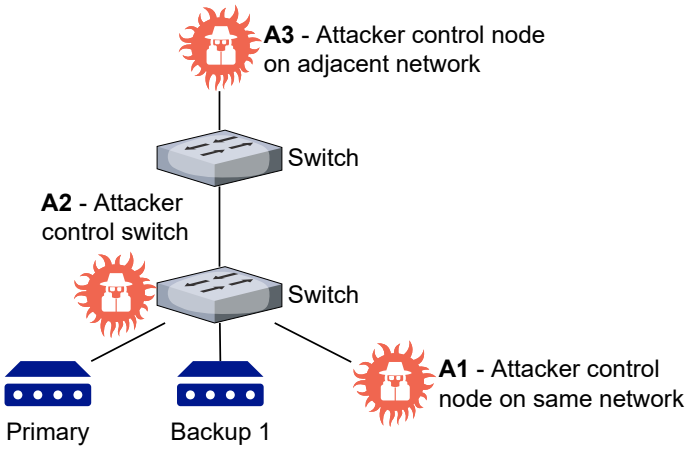


Fig. 3: Attacker model.

The threat model above is assuming all kinds of attacks are equally exploitable, which is of course not the case. In the following we categorize the attackers, and describe which threats each category could possibly exploit.

In the analysis, we have omitted the attacker category with direct physical access to equipment. Unless extremely sophisticated, that category would not be able to perform other attacks than direct physical harm to the control equipment or network infrastructure, i.e., kinetic DoS attacks. The more sophisticated attacks include, e.g., the ability to attach to physical debug ports or attaching rogue equipment to pins of boards. Such attacks can have a severe impact, but they are specific to the used hardware and firmware and is therefore not handled in this analysis which is done on a more generic level.

Let us assume an attacker model, where the attacker falls into one of the following classes, illustrated in Fig. 3:

A1 Controls a device on the same network as the redundant set of nodes.

A2 Controls a node that sits between the primary and redundant device, i.e., any communication between backup and primary must pass through this node. For example a network switch or A1 after a successful ARP-spoofing attack.

A3 Controls a device on an adjacent network.

The relationship between the threats and attacker classes is provided in the Attacker column of Table I. The column indicates which classes of attackers could possibly exploit a vulnerability related to the given threat.

C. Data flow diagram and interactions of interest

A data flow diagram of the redundancy links are depicted in Fig. 2. There are only three interactions crossing trust boundaries in the DFD, P1:DF1:P2, P1:DF2:P2 and P2:DF1:P1, which are analyzed in subsequent sections.

D. Threat analysis

Each threat is categorized according to STRIDE, given an ID, associated with potential threat consequences as well as which attackers can potentially be threat actors, and which methods could be used to mitigate the threat risk. All this is summarized in Table I.

Non-repudiation threats arise if security relevant events occur without traceability that, e.g., a change in a firewall configuration which cannot be traced back to the user performing the configuration. Sending heartbeat-data as well as state is not security relevant events, which means this threat is inapplicable and therefore not considered in the threat model.

Heartbeat from primary to backup (P1:DF1:P2 in Fig. 2): The heartbeat data sent from the primary to the backup are used to inform the backup entity on the health of the primary, and is sent out with a given frequency. In case the backup does not receive any heartbeat messages, or there are indications in the message that the primary has a deteriorated health, the backup will initiate a role selection round.

An attacker successfully spoofing, replaying or tampering with the heartbeat data in (threats 1-3) can trick the backup to initiate a role selection round, resulting in dual primary (TC1), or loss of redundancy (TC3) in a worst case scenario. An elevation of privilege attack abuses weak access control mechanisms to gain access to resources it should not be privileged to use. In this case this means that an authentic member of the system is able to send heart-beat data to the backup unit(s) without being the current primary. Similar consequences as for threats 1-3 can be seen for threat 8. It could also be possible for the attacker to force a switchover even though it not is needed (TC8). An iterative series of such switchovers would generate a system that look healthy from the outside, but in which the start-up behavior will render the system effectively non-functional.

Any of the threats 1,2 and 8 would have less predictable consequences in case P1 is still sending heartbeat messages, meaning a mix of genuine and crafted/replayed heartbeat messages reach the backup. Therefore, an attacker succeeding in effectuating threat 5 in combination with any of the threats 1, 2 or 8 would have a considerably higher chance of achieving its goals.

Information disclosure threats occur if sensitive information is leaked to unauthorized parties. It would be possible to eavesdrop on the heart-beat messages (threat 4) for an attacker sitting between the redundant entities, resulting in consequence TC6.

The DoS threats are split in three. They have similar consequences, but are executed in different ways. Hindering the heartbeat data to reach the backup (threat 5) requires control of an intermediate node between P1 and P2. Spamming the network interface of the backup with fake heartbeat data could lead to the internal heartbeat processing mechanism within the backup to work in a deteriorated state (threat 6). Overloading the network interface of the backup or of an intermediate switch (threat 7) could lead to genuine heartbeat messages potentially being dropped. All of these threats could lead to loss of redundancy (TC3), and in some cases to dual primaries (TC1).

State from primary to backup (P1:DF2:P2 in Fig. 2): The state transfer link is used to transfer application specific state-data from the primary service to the backup, so that the backup can transition into being primary with a minimum delay. Size of a full state transfer is application specific, but typically cannot fit into one telegram, therefore a stream of telegrams is considered for a full state transfer.

Successfully spoofing or replaying state data (threat 9, 10), as well as tampering with state data content (threat 11) could lead to a backup controller to enter a state of the attackers choice, provided that the attacker has sufficient knowledge about the application characteristics running in the redundant set. The same type of consequence can be seen for EoP (threat 14). Similarly as for the previous link (heartbeat to backup), the consequences are difficult to foresee in case the genuine primary is still sending state messages. For a predictable outcome, an adversary would need to combine the spoofing attack with a DoS attack on the secondary link.

Information disclosure of state data (threat 12) would lead to consequence TC4.

Different variants of DoS threats against the state transfer link (threat 13-15) include the same set as described for the heartbeat link to backup. The consequence in this case would mainly be in loss of redundancy TC3, as the internal state handling within the backup will lower its health in case the state is not periodically updated. In practice it is also TC5, as the primary and backup view of the state will deviate after lost state transfer.

The major consequence for the threats against the state transfer link is that the state of the backup deviates from the state in the primary TC5. On a switchover, the backup taking over starts setting I/O signals according to its (potentially erroneous) internal state, leading to consequence TC7, which is a potential threat to the safety of the system, and thereby the most severe consequence. This however requires a very elabo-

TABLE I: Condensed overview of the security analysis.

Interaction	Category	ID	Description	Consequence	Attacker	Mitigation
P1:DF1:P2 Heartbeat from primary to backup	S	1	Attacker sending crafted heartbeat messages.	TC1, TC3	A1, A2, A3	M1, M5
	S	2	Replay of message sniffed by an attacker.	TC1, TC3	A1, A2	M3
	T	3	Altering heartbeat message.	TC1, TC3	A2	M1
	R	-	Not applicable	-	-	-
	I	4	Disclosure of heartbeat data	TC6	A1, A2	M2
	D	5	Cutting heartbeat traffic.	TC1, TC3	A2	M7, M6
	D	6	Overload of heartbeat data.	TC1, TC3	A1, A2, A3	M8, M9, M5
	D	7	Overload of switch/network interface.	TC1, TC3	A1, A2, A3	M8, M9, M5
	E	8	Unauthorized user send heartbeat to backup.	TC1, TC3, TC8	A1, A3	M4
P1:DF2:P2 State from primary to backup	S	9	Attacker sending crafted state data.	TC5	A1, A2, A3	M1, M5
	S	10	Replay of message sniffed by an attacker.	TC5	A1, A2	M3
	T	11	Intercepting and changing the content of genuine state messages.	TC5	A2	M1
	R	-	Not applicable	-	-	-
	I	12	Disclosure of state data	TC4	A1, A2	M2
	D	13	Cutting state traffic.	TC3, TC5	A1, A2, A3	M6, M7, M5
	D	14	Overload of state data.	TC3, TC5	A1, A2, A3	M8, M9, M5
	D	15	Overload of switch/network interface.	TC3, TC5	A1, A2, A3	M8, M9, M5
	E	16	Unauthorized user sends state to the backup.	TC5	A1, A3	M4
P2:DF1:P1 Heartbeat from backup to primary	S	17	Attacker sending crafted heartbeats.	TC2, TC3, TC8	A1, A2, A3	M1, M5, M3
	S	18	Replay of message sniffed by an attacker.	TC2, TC3, TC8	A1, A2	M3
	T	19	Altering heartbeat message.	TC2, TC3, TC8	A2	M1, M5
	R	-	Not applicable	-	-	-
	I	20	Disclosure of heartbeat data	TC6	A1, A2	M2
	D	21	Cutting heartbeat traffic.	TC1, TC2, TC3, TC8	A2	M7, M6
	D	22	Overload of heartbeat data.	TC2, TC3	A1, A2, A3	M8, M9, M5
	D	23	Overload of switch/network interface.	TC2, TC3	A1, A2, A3	M8, M9, M5
	E	24	Unauthorized user sends heartbeat to primary.	TC2, TC3, TC8	A1, A3	M4

rate type of attack, which is able to successfully alter the state of the backup unit and then initiate a successful switchover.

Heartbeat from backup to primary (P2:DF1:P1 in Fig. 2): In case of a backup receiving an indication on deteriorated health of the primary, a role selection protocol is started, where all units in the redundancy set send role-selection heartbeats, witch ends with the unit with best health status taking the role as the new primary.

The threats against the heartbeat for role selection are very similar to the ones for primary health monitoring, but the consequences differ. Furthermore, the threat consequences are more predictable, because a backup unit only sends this type of data to the primary on role selection. Potential consequences are no primary (TC2), loss of redundancy (TC3) and unnecessary switchover (TC8), which are shared between the spoofing, replay, tampering and EoP threats (17-19, 24).

The Information disclosure of this type of role-selection heartbeat (threat 20) may seem of limited interest, but it indicates that a switchover is imminent (TC6), which may be a trigger for other types of attacks.

Hindering a role selection heartbeat from reaching the primary (threats 21-23), could potentially lead to dual primary (TC1), no primary TC2, or loss of redundancy (TC3) depending on which message in the sequence is lost.

E. Mitigation enumeration

For the different threats there are several well known mitigation mechanisms, some on protocol level, some on the network architecture, and some with the help of external components.

Protocol threat mitigation mechanisms:

M1 *Integrity and authenticity*: Mechanisms for protecting the integrity and authenticity of the data are required to counter tampering and spoofing threats. Integrity means that data cannot be altered, authenticity means that the source of the data can be validated. Checksum such as CRCs and data signing using e.g., Hashed Message Authentication Codes (HMAC) [28] are typical methods providing such capabilities.

M2 *Encryption*: Point-to-point encryption of sensitive data in transit utilizing, e.g., IPSec or TLS [28].

M3 *Session handling*: Communication using proper session handling will mitigate replay attacks, if session IDs are renewed sufficiently often. Other freshness indications, such as a timestamp or sequence number, could be used if session-less communication is needed.

M4 *Access control*: Describing and enforcing sufficiently fine-grained policies following the principle of least privilege [29] is the state of the art for access control. For the redundancy links, this could mean that only the configured redundant partners are allowed to open sessions for transferring the heartbeat or state data.

Network architectural threat mitigations:

M5 *Restricted Data flow*: Disallowing or limiting routing between network segments is one way of mitigating attacks coming from external threats, e.g., from A3. Network isolation is the extreme of this mitigation method, a quite commonly used strategy in IACS.

M6 *Network redundancy*: Using a network redundancy protocol such as PRP could increase the network availability, making some of the DoS threats more difficult to effectuate.

M7 *Link replication*: Similarly, replicating traffic on several links would mitigate against DoS threats. Particularly, sending heartbeat on one link and state on another would make attack chains which rely on exploiting threats on both of these links more difficult to perform.

Threat mitigations through external components:

M8 *Device firewalls*: Firewalls and similar local network hardening mechanisms can protect against some of the DoS style attacks, including overload attacks.

M9 *Switch firewalls*: Firewalls and similar capabilities of the network infrastructure can protect against DoS attacks.

Each of these high-level mitigation mechanisms are associated with respective threat in Table I.

VI. Discussion

A security analysis is to be seen as a starting point in the process of implementing a sufficiently secure solution, in this case for controller redundancy. The analysis helps in making informed decisions on what mitigation mechanisms to implement or integrate, based on knowledge of the potential weaknesses of the selected design. The analysis should also be used as an input to the security validation and verification plan to ensure that the selected mechanisms are properly implemented and provides the needed protection.

From the analysis it is clear that attacks on the heartbeat link as well as the state transfer link could have sever impact on essential functionality of a controller. Loss of safety is possible if a threat actor can effectuate threats on the state transfer link to put the backup in a state of the attackers choice, and then on the heartbeat link to force an unnecessary switchover. This attack is however very elaborate and requires a highly motivated and knowledgeable attacker.

Providing heartbeat and state data with basic integrity and authenticity protection is a very effective protocol level protection, mitigating threats related to spoofing and tampering. Encryption is also a good option, even though the value for the heartbeat link may be limited as it will only counter threat consequences rated *Low*. Encryption of the state data link should be considered, as the content of the state being transferred is application specific and therefore potentially sensitive. Providing the system owner with an option to enable encryption on the state data link could be a good compromise, as the real-time performance of state transfer potentially deteriorates by encrypting the state data.

Session handling (or at least some sufficiently robust freshness indication) of heartbeat as well as state data is required to counter replay attacks. Session based protocols also provide a number of additional benefits such as retransmission and message acknowledge, potentially at the cost of an increased latency.

Implementing access control for the heartbeat and state data links will protect against the effect of a misconfiguration as well as against insider threats, both of which the other mechanisms cannot easily mitigate.

The threat mitigation mechanisms related to the network architecture and firewalls are effective against threats in the DoS category. Using the total network isolation described as the extreme of M5 will mitigate any external threats, but is generally untenable in NCC. Notably, M7 will limit the effects of attacks on the links sending cyclic data, e.g., the cyclic health monitoring heartbeat of the primary, as long as the replicated links are still alive. This mitigation will, however, come with additional hardware and software requirements, resulting in a more complex solution which may contain additional attack paths.

The completed security analysis is only partial, it lacks the analysis of residual risk after mitigation, as the model is done without a specific protocol implementation or architecture in place. Therefore a full analysis cannot be completed. As such, the provided analysis can help in giving guidance on what mechanisms are needed, and what the potential consequences could be - doing nothing with regards to securing redundancy links is obviously a bad option. There are several communication protocols available that provide several of the described mitigating mechanisms, such as OPC UA [30], TLS and DDS[2]. However, the selection of protocol, architecture and compensating countermeasures will impact the performance

[2]www.dds-foundation.org

of the redundancy links, and thereby the switchover time, implying the selection of mechanisms must be done with care.

VII. Conclusions

This study provides a security analysis of the links required for warm standby controller redundancy for network-centric control. This is a vital aspect of the evolving flexible design strategies within the automation industry. The approach for analysis is an extended method for threat-modeling including threat consequence enumeration, attacker category enumeration and mitigation mechanisms. The results underlines the importance of securing the data links used for redundancy data transfer when transitioning into network-centric control architectures.

In the analysis, 24 different threats are enumerated, described and associated with threat consequences, attacker category and possible mitigating measures. Several of these threats have the potential to cause severe consequences, including loss of essential functionality.

Redundancy switchover is often time-critical. The maximum time after which a backup can take over as primary must be a known factor during engineering. As future work we would like to experiment on a select set of security enhancing protocols to study performance impact on the redundancy links and switchover time. After mitigating measures are selected, the analysis of residual risks can be done, to complement this study.

The threat model analysis hypothesizes over a set of potential attacks of which many are seemingly difficult to realize. An interesting future challenge would be to try to implement these attacks in a realistic system, to understand the required conditions and set of knowledge needed by an attacker to be successful. This could provide insight in additional security measures or weaknesses not uncovered by the theoretical analysis of this paper.

Acknowledgements

We would like to thank Emmy Zhou and Volodmyr Trykoz for help with the language review. The work is supported by ABB AB and the industrial postgraduate school Automation Region Research Academy (ARRAY) and SEINE project, both funded by The Swedish Knowledge Foundation (KKS).

References

[1] A. Simion and C. Bira, "A review of redundancy in plc-based systems," *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies XI*, vol. 12493, pp. 269–276, 2023.

[2] "IEC 62439-3 Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)," IEC, Geneva, CH, Standard, 2021.

[3] "IEC 62439-2 Industrial communication networks - High availability automation networks - Part 2: Media Redundancy Protocol (MRP)," IEC, Geneva, CH, Standard.

[4] N. Budhiraja, K. Marzullo, F. B. Schneider, and S. Toueg, "The primary-backup approach," *Distributed systems*, vol. 2, pp. 199–216, 1993.

[5] PACSys, "PACSystems™ RX3i Hot Standby CPU Redundancy," https://emerson-mas.my.site.com/communities/en_US/Documentation/PACSystems-Hot-Standby-CPU-Redundancy-Users-Manual, 2023, accessed: 2024-04-08.

[6] B. Leander, B. Johansson, T. Lindström, O. Holmgren, T. Nolte, and A. V. Papadopoulos, "Dependability and security aspects of network-centric control," in *28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2023, pp. 1–8.

[7] Fortinet, "1H 2022 Global Threat Landscape Report," Tech. Rep. 1, 2022.

[8] M. Iaiani, A. Tugnoli, S. Bonvicini, and V. Cozzani, "Analysis of cybersecurity-related incidents in the process industry," *Reliability Engineering & System Safety*, vol. 209, p. 107485, 2021.

[9] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," NIST, Gaithersburg, MD, Tech. Rep., 2020.

[10] S. Singh, V. M. Chary, and P. A. Rahman, "Dual redundant profibus network architecture in hot standby fault tolerant control systems," in *Int. Conf. on Advances in Eng. & Tech. Research (ICAETR)*, 2014.

[11] J. Stój, "Cost-effective hot-standby redundancy with synchronization using ethercat and real-time ethernet protocols," *IEEE Trans. on Autom. Science and Eng.*, vol. 18, no. 4, pp. 2035–2047, 2020.

[12] T. Hegazy and M. Hefeeda, "Industrial automation as a cloud service," *IEEE Trans. Par. and Distr. Syst.*, vol. 26, no. 10, pp. 2750–2763, 2015.

[13] B. Johansson, M. Rågberger, T. Nolte, and A. V. Papadopoulos, "Kubernetes orchestration of high availability distributed control systems," in *ICIT*, 2022.

[14] I. Álvarez, A. Ballesteros, M. Barranco, D. Gessner, S. Djerasevic, and J. Proenza, "Fault tolerance in highly reliable ethernet-based industrial systems," *Proc. IEEE*, vol. 107, no. 6, pp. 977–1010, 2019.

[15] P. Danielis *et al.*, "Survey on real-time communication via ethernet in industrial automation environments," in *19th International Conference on Emerging Technology and Factory Automation (ETFA)*. IEEE, 2014.

[16] "IEC 62443 part 4-1: Security for industrial automation and control systems - secure product development lifecycle requirements," IEC, Geneva, CH, Standard, 2009-2018.

[17] B. Leander, A. Čaušević, and H. Hansson, "Applicability of the IEC 62443 standard in Industry 4.0/IIoT," in *14th International Conference on Availability, Reliability and Security (ARES)*. ACM, 2019.

[18] R. Khan, K. McLaughlin, D. Laverty, and S. Sezer, "STRIDE-based threat modeling for cyber-physical systems," in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*. IEEE, sep 2017, pp. 1–6.

[19] B. Leander, A. Čaušević, and H. Hansson, "Cybersecurity challenges in large industrial iot systems," in *24th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 1035–1042.

[20] D. H. Shin, G. Y. Kim, and I. C. Euom, "Vulnerabilities of the Open Platform Communication Unified Architecture Protocol in Industrial Internet of Things Operation," *Sensors*, vol. 22, no. 17, 2022.

[21] R. Ma *et al.*, "Stealthy Attack Against Redundant Controller Architecture of Industrial Cyber-Physical System," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9783–9793, 2019.

[22] C. Preschern, N. Kajtazovic, and C. Kreiner, "Built-in security enhancements for the 1oo2 safety architecture," in *Proceedings - 2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems, CYBER 2012*. IEEE, 2012, pp. 103–108.

[23] H. Sepehrzadeh, "Security Evaluation of Cyber-Physical Systems with Redundant Components," in *Proceedings - 2022 CPSSI 4th International Symposium on Real-Time and Embedded Systems and Technologies, RTEST 2022*. IEEE, 2022, pp. 1–7.

[24] B. Johansson, M. Rågberger, A. V. Papadopoulos, and T. Nolte, "Consistency before availability: Network reference point based failure detection for controller redundancy," in *28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2023, pp. 1–8.

[25] B. Johansson, M. Rågberger, A. V. Papadopoulos, and T. Nolte, "Heartbeat bully: failure detection and redundancy role selection for network-centric controller," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2020, pp. 2126–2133.

[26] M. Howard and S. Lipner, *The security development lifecycle*. Microsoft Press Redmond, 2006, vol. 8.

[27] Rebecca M. Blank. Patrick D. Gallagher, "NIST Special Publication 800-30 Revision 1 - Guide for Conducting Risk Assessments," Tech. Rep. September, 2012.

[28] R. Shirley, "Internet Security Glossary, Version 2," RFC 4949, Aug 2007.

[29] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.

[30] "IEC 62541 OPC unified architecture," IEC, Geneva, CH, Standard, 2016.